



Centro Universitário de Brasília – UniCEUB
Faculdade de Tecnologia e Ciências Sociais Aplicadas - FATECS

Eduardo Morici Gonçalves

Implementação de Alta disponibilidade em Máquinas Virtuais utilizando Software Livre

Brasília
2009

Eduardo Morici Gonçalves

Implementação de Alta disponibilidade em Máquinas Virtuais utilizando Software Livre

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB/FATECS) como pré-requisito
para a obtenção de Certificado de
Conclusão de Curso de Graduação no curso
de Engenharia da Computação.
Orientador: Prof. Dr. Miguel Archanjo
Bacellar Goes Telles Junior

Brasília
2009

Eduardo Morici Gonçalves

Implementação de Alta disponibilidade em Máquinas Virtuais utilizando Software Livre

Trabalho apresentado ao Centro
Universitário de Brasília
(UniCEUB/FATECS) como pré-requisito
para a obtenção de Certificado de
Conclusão de Curso de Graduação no curso
de Engenharia da Computação.

Orientador: Prof. Dr. Miguel Archanjo
Bacellar Goes Telles Junior

Brasília, ____ de julho de 2009

Banca Examinadora

Prof. Dr.

Prof. Dr.

Prof. Dr.

Prof. Dr.

Agradecimento

Os grandes momentos da vida são construídos, aos poucos, através de muito suor, sofrimento e alegrias.

Agradeço primeiramente a Deus, o Grande Mestre criador, que nos deu a vida e a chance de, a partir de muito trabalho e dedicação, alcançar aquilo que almejamos.

Agradeço também aos meus pais, Marcio e Marilda, que me trouxeram ao mundo, me educaram e me conduziram ao que me tornei hoje. Agradeço também a minha mulher, Fernanda, mulher de grande fibra e moral, a qual eu escolhi para compartilhar sonhos e realizações.

Agradeço ainda aos professores do UniCEUB que me transmitiram conhecimento e ajudaram na minha formação profissional e em especial ao professor Miguel Archanjo, que me orientou nesse trabalho de conclusão de curso.

Agradeço ainda a alguns amigos que me auxiliaram também neste momento, sem mesmo ter qualquer tipo de vínculo ou recompensa com este trabalho. São eles Henri Dario Rugel, Jailton Caldeira (Bicudo) e Edimilson Silva (Pajé).

Não há como deixar de agradecer a um grande amigo, pessoa de minha total confiança que me observou e auxiliou em todo esse trabalho, José Gabriel da Costa.

A todos vocês, sou grato.

“If you don` t have walls, you don` t need Windows.”

Autor Anônimo

Resumo

A virtualização de servidores está cada dia mais sendo utilizada em universidades e empresas, sejam elas empresas públicas, datacenters ou até mesmo empresas que nada tem a ver com informática. A busca pela utilização da tecnologia de virtualização, na grande maioria das vezes, é a busca pela redução do custo da solução adquirida.

Porém, sistemas de virtualização são caros e de administração complexa, o que gera um custo irreal para 98% das empresas brasileiras, como citado pelo Sr Rodolfo Rodrigues Soares em seu Trabalho de conclusão de curso. Existe outro ponto que também entra nesta pauta de discussão, que é a disponibilidade. Sistemas virtualizados costumam apresentar baixa disponibilidade, o que inviabiliza sua aquisição. A migração de máquinas virtuais utilizando software livre visa aumentar a disponibilidade do ambiente e viabilizar a utilização deste em empresas de todos os portes.

Palavras chaves: Virtualização, migração, redundância, escalabilidade, confiabilidade, disponibilidade.

Lista de Tabelas

Tabela 1	Classificação de disponibilidade	Pag. 27
Tabela 2	Configuração dos equipamentos que compõem o cluster	Pag. 41
Tabela 3	Procedimento de configuração do particionador.	Pag. 47
Tabela 4	Plano de teste 1	Pag. 58
Tabela 5	Resultado do teste de indisponibilidade	Pag. 59
Tabela 6	Plano de teste 2	Pag. 60
Tabela 7	Resultado do teste de latência	Pag. 61
Tabela 8	Plano de teste 3	Pag. 62
Tabela 9	Resultado do teste de desempenho	Pag. 63
Tabela 10	Análise de TCO	Pag. 65

Lista de Figuras

Figura 1	Particionamento baseado em anéis de privilégios, padrão do MULTICS	Pag. 22
Figura 2	Arquitetura de Servidor com virtualização baseada em VMM sobre Sistema Operacional	Pag. 24
Figura 3	Arquitetura de Servidor com virtualização instalada direta no Hardware.	Pag. 25
Figura 4	Estágios do processo de Migração de Máquinas Virtuais	Pag. 33
Figura 5	Topologia Física adotada entre os servidores	Pag. 40
Figura 6	Switch Linksys WRT54G utilizado no projeto	Pag. 40
Figura 7	Servidor de Virtualização e Banco de dados	Pag. 41
Figura 8	Topologia lógica da solução apresentada	Pag. 42
Figura 9	Tela de configuração do Particionador	Pag. 47
Figura 10	Estrutura de divisão do Disco Rígido.	Pag. 49
Figura 11	Destino iSCSI através do configurador YAST	Pag. 52
Figura 12	Iniciador iSCSI através do configurador YAST	Pag 53

Sumário

Agradecimento	iv
Resumo	vi
Lista de Tabelas	vii
Lista de Figuras	viii
Sumário	ix
Lista de Termos e Abreviaturas	xi
1. Introdução	13
1.1 Motivação.....	14
1.2 Objetivo Geral.....	15
1.3 Objetivos Específicos.....	16
1.4 Organização dos capítulos	17
2 Virtualização e Alta disponibilidade.....	19
2.1 Virtualização de sistemas operacionais.....	19
2.2 Objetivos da Virtualização.....	19
2.3 Como funciona a virtualização	21
2.4 Definido alta disponibilidade.....	25
2.5 Formas de aferição de disponibilidade de sistemas	27
2.6 Parâmetros de medida de disponibilidade.....	28
3 Virtualização com Software Livre Utilizando XEN	30
3.1 Sobre o XEN	30
3.2 Formas de virtualização no XEN	31
3.2.1 Virtualização Completa.....	31
3.2.2 Paravirtualização	32
3.3 Ferramentas de migração de servidores em tempo de execução – Máquinas Vivas	32
3.4 Ferramentas de alta disponibilidade.....	35
3.4.1 OCFS2 - sistema de arquivo compartilhado	35
3.4.2 Heart Beat2 – Monitoramento do cluster.....	36
3.4.3 iSCSI – Internet Small Computer System Interface	37
4 Instalando o Ambiente Virtualizado.	39
4.1 Topologia da Solução.....	39
4.1.1 Topologia Física.....	39
4.1.2 Topologia Lógica	42
4.2 Instalação do Sistema Operacional para Ambientes Virtualizados	43
4.3 Configuração do ambiente para funcionamento	44
4.3.1 Configurando o Sistema Operacional	44
4.3.2 Configuração do NTP	45
4.3.3 Configuração do arquivo hosts	45
4.3.4 Configuração do Particionamento do disco	46
4.3.5 Configurando o Cluster.....	50
4.3.5.1 Configuração do OCFS2.....	50
4.3.5.2 Configuração do iSCSI	51
4.3.5.2.1 Configuração do Destino iSCSI.....	51
4.3.5.3 Configuração do Iniciador iSCSI.....	52
4.3.6 Configuração do Sistema de Migração de Máquinas Vivas	55
4.3.7 Outras configurações gerais no sistema	56

5	Testes	57
5.1	Teste de disponibilidade.....	57
5.1.1	Análise de tempo de indisponibilidade	57
5.1.2	Análise de latência	59
5.2	Análise de desempenho.....	62
6	Conclusão.....	64
6.1	Projetos Futuros	65
	Referências.....	67
	Anexo I – Proposta comercial Star Company do Brasil	69
	Anexo II – Nota Fiscal de aquisição de equipamentos	71

Lista de Termos e Abreviaturas

100BASE-TX	Padrão conhecido como Fast Ethernet
AMD	Advanced Micro Devices
Bare Metal	Sistema de virtualização instalado direto no hardware, sem sistema operacional
Benchmark	Busca das melhores práticas na indústria que conduzem ao desempenho superior
Binário	Número com base 2
BSD	Berkeley Software Distribution
Bytes	1 Byte é igual a 8 bits
Centrino	Nome comercial da Intel para definir modelo de processadores para notebooks
CITRIX	Empresa americana do ramo de virtualização
Cluster	Aglomerado de computadores
Data Centers	Centro de processamento de dados
Datagrama	É a estrutura de dados unitária de transmissão em uma rede (de computadores ou telecomunicações).
Down Sizing	Racionalização de custos. Diminuição dos gastos com equipamentos de grande porte e aquisição de equipamentos de menor porte.
E/S	Este termo é utilizado quase que exclusivamente no ramo da computação (ou informática), indicando entrada (inserção) de dados por meio de algum código ou programa, para algum outro programa ou hardware, bem como a sua saída (obtenção de dados) ou retorno de dados, como resultado de alguma operação de algum programa, conseqüentemente resultado de algum input.
Fast ethernet	Rede de dados com transmissão a 10/100 Mbps.
FC	Fiber Channel
FreeBSD	sistema operacional livre do tipo Unix descendente do BSD desenvolvido pela Universidade de Berkeley
FTP	File Transfer Protocol
Full Virtualization	Virtualização Completa
GNU	GNU is Not Unix
Hardware	Parte física de um equipamento de informática
HD	Hard Drive
HP	Hewlett-Packard
IBM	International Business Machines
Inodes	Estrutura responsável por conter informações básicas sobre arquivos e pastas, como permissões de acesso, data e hora do último acesso, tamanho e ponteiros para o arquivo
Intel Celeron	Processadores de baixo custo desenvolvido pela Intel Corporation para equipamentos residenciais.
IP	Internet Protocol
Journaling	Um sistema de arquivos com journaling dá permissão ao Sistema Operacional de manter um log (journal), de todas as mudanças no sistema de arquivos antes de escrever os dados no disco. Normalmente este log é um log circular alocado em uma área especial do sistema de arquivos.
Kbytes	1 Kbyte é igual a 1024 bytes
Keepalive	Sinal enviado frequentemente em intervalos predefinidos. Após um

	sinal ser enviado, se não for recebida resposta é presumido que o link está fora e futuros dados serão encaminhados através de outro caminho até que o link esteja de novo funcional.
Kernel	O Kernel de um sistema operacional é entendido como o núcleo deste ou, numa tradução literal, cerne.
Linux	Kernel e Sistema Operacional desenvolvido por Linux Torvalds
Lock	Bloqueio de escrita em arquivo de dados
LUN	Logical Unit – Unidade Lógica
Main Frame	Computador de grande porte, dedicado normalmente ao processamento de um volume grande de informações
micro-instruções	Instruções escritas dentro do processador
MTBF	Mean Time Between Failures
MTTR	Mean Time to Repair
MULTICS	Multiplexed Information and Computing Service
NetBSD	NetBSD é um sistema operacional de código aberto que busca ser seguro e portátil para diversas plataformas.
NetOS	Sistema operacional desenvolvido pela Universidade da Califórnia
NTP	Network Time Protocol
Oracle	Empresa americana especialista no desenvolvimento de Bancos de Dados
RAM	Random Access Memory
Red Hat	A Red Hat, Inc. é uma empresa dos Estados Unidos, que disponibiliza soluções baseadas no sistema operativo GNU/Linux, incluindo várias distribuições.
SCSI	Small Computer System Interface
Sistemas Hospedes	Sistemas operacionais instalados sobre Sistemas de Virtualização como XEN ou VMWare
Storage Área Networking	Na computação, um Storage Area Network (área de armazenamento em rede) ou SAN é uma rede projetada para agrupar dispositivos de armazenamento de computador. Os SANs são mais comuns nos armazenamentos de grande porte.
TCO	Total Coast Ownership – Custo Total de Propriedade
TCP	Transmission Control Protocol
Timeout	Tempo esgotado
UDP	User Datagram Protocol
UTP	Unshielded Twisted Pair
VMM	Virtual Machine Manager
Writeback	Processo de retorno no processo incompleto de escrita no caso de falha do procedimento de escrita de arquivos.
x86	Em informática, x86 ou 80x86 é o nome genérico dada à família (arquitetura) de processadores baseados no Intel 8086, da Intel Corporation.
XEN	Xen é um software livre de virtualização para as arquiteturas x86, x86-64, IA-32, IA-64 e PowerPC.
Yast	Yet another Setup Tool – Ferramenta de configuração padrão do OpenSuse.

1. Introdução

Virtualização é um dos termos forjados na tecnologia que ganham fama como sendo a panacéia, a cura de todos os males da tecnologia, uma tecnologia que certamente acarretará em redução drástica de custos e principalmente em uma consolidação fácil de servidores.

A tecnologia de virtualização de recursos computacionais nasceu a mais de 3 décadas, juntamente ao princípio da computação comercial, nos anos de 1960, com a idéia de Compartilhamento de Tempo (*Time Sharing*), onde o professor da universidade de *Oxford*, Christopher Strachey, lançou uma publicação científica chamada “*Time Sharing in Large Fast Computers*”. Hoje é utilizada em quase todas as vertentes da tecnologia. Termos como VPN (*Virtual Private Network*) e VTL (*Virtual Tape Library*) são alguns exemplos de como o termo virtualização ganhou a grande uso em todos os campos da tecnologia, e continua sendo muito utilizado neste meio. [WILLIANS; GARCIA, 2007]

A forma mais adotada de virtualização de recursos computacionais é a utilização de máquinas virtuais emulando ambientes de máquinas físicas para compartilhar recursos computacionais. Essa adoção é adotada hoje na divisão dos recursos de hardware, utilizando uma camada de abstração entre o Hardware e o Sistema Operacional que possibilita o compartilhamento desses recursos entre diversos sistemas operacionais instalados no mesmo equipamento. [MARSHALL; RAYNOLDS, 2006]

Com isso, a utilização de máquinas virtuais é hoje uma tendência crescente, como mostrou o Gartner Group, desde 2004, a virtualização vem crescendo e fazendo parte da maioria das empresas *Fortune 1000* americanas, e

também das demais empresas no mundo. Isto torna a virtualização um ponto de interesse do mercado de tecnologia, justificando investimentos neste tipo de aplicação e uma busca do mercado às aplicações que possibilitam a virtualização dos recursos computacionais, com a manutenção da disponibilidade desses recursos compartilhados. [BITTMAN, 2003]

1.1 Motivação

A motivação ao uso de sistemas virtualizados é a redução global do Custo Total de Propriedade (*TCO*), quando comparada a uma solução completamente implementada sobre *hardwares*, com sistema operacional dedicado.

Porém, existe um ponto de falha na simples utilização de um *hardware* virtualizado, que é a importância de um ambiente utilizando sistema de virtualização computacional, onde, se em uma máquina, rodando apenas um serviço, há uma falha de hardware ou software, apenas aquele serviço será prejudicado. Por outro lado, se há a queda de um sistema de virtualização de recursos computacionais com, por exemplo, cinco sistemas operacionais virtualizados sobre ele, compartilhando recursos, o dano causado é potencializado em, pelo menos, cinco vezes. Desta mesma forma, caso seja utilizada uma máquina de altíssimo desempenho, com capacidade de virtualizar até 128 máquinas virtuais, a indisponibilização desse sistema causaria uma perda potencialmente irreparável para a empresa. [SOARES, 2006] e [VMWARE, 2006]

A utilização de recursos de alta disponibilidade hoje é uma vertente forte em todo o segmento de virtualização, possibilitando a migração de máquinas virtuais em estado de execução, manipulando de tal forma os estados de conexão e a paginação da memória que, se o processo ocorrer em um prazo de até 60 ms torna-se imperceptível para os usuários do serviço, não sendo necessária nem mesmo a reconexão de rede das estações cliente. [CLARK; FRASER, 2005]

Hoje existem sistemas que possuem soluções e alta disponibilidade e que mitigam o risco em ambiente virtualizados, como por exemplo, o VMWare ESXi, software comercializado a um preço alto, que, de acordo com cotação disponibilizada em novembro de 2008, pode ter seu preço, por cada quatro processadores, estimado em R\$ 45.000,00, conforme o Anexo I, cotação de preço feita junto a empresa Star Company do Brasil.

Essas versões pagas de sistemas de Virtualização, como exemplificado acima, são caras e não condizem com a realidade das empresas brasileiras, que até o ano 2000 se constituíam 98% somente por microempresas, empresas essas que possuem seu nível de faturamento anual de até R\$ 1.000.000,00 (um milhão de Reais). Essa realidade de faturamento impossibilita a tais empresas a aquisição de recursos de tecnologia de virtualização que possuem recursos de alta disponibilidade, nos formatos comerciais padrão, ou seja, de forma paga. [SOARES, 2006]

1.2 Objetivo Geral

Este projeto implementa máquinas virtuais em ambiente de alta disponibilidade, montados em ambientes de baixo custo, se tornando uma solução

estável e disponível. Para tanto será utilizado software livre e hardware de uso comum, sem nenhuma especialidade, utilizando computadores com arquitetura de processadores CISC de baixo custo, ou seja *Intel Celeron*, que foram adquiridos em um supermercado de Brasília, mostrando que é possível implementar uma solução para pequenas empresas e corporações.

1.3 Objetivos Específicos

Como objetivo específico, vamos analisar o ambiente em seu funcionamento e seu custo, com os seguintes índices de referência.

- TCO da solução

Para análise de TCO, é verificada a real viabilidade da solução do ponto de vista financeiro, demonstrando se o custo proposto na solução é viável para uma empresa de porte pequeno no Brasil.

- Grau de estabilidade

Para análise de estabilidade, é verificado se a solução é funcional com o menor período de tempo de indisponibilidade, em caso de necessidade de alteração de servidor de virtualização, e também o aumento da latência nas transações durante o processo de migração de máquinas virtuais.

- Grau de disponibilidade

Para análise de disponibilidade, é verificado se o tempo de recuperação da solução em caso de falha de um dos nós do *cluster*, restaurando a máquina para seu estado funcional. Nessa análise, leva-se em conta paradas programadas e não programadas.

- Facilidade de implementação da solução.

Para análise de facilidade de implementação, leva-se em conta o grau de alteração que deve ser feito no ambiente para que se obtenha disponibilidade nas máquinas virtuais.

1.4 Organização dos capítulos

2. Virtualização e Alta Disponibilidade

Este capítulo trata da teoria de virtualização, suas particularidades e funcionamento. Considera também as vantagens trazidas pela virtualização de sistemas operacionais. Essas vantagens são definitivas em ambientes virtualizados.

Este capítulo trata também do conceito de alta disponibilidade, definição do termo e formas de medição de disponibilidades de sistemas. Trata ainda da utilização de alta disponibilidade em ambiente virtualizados, como é possível e o que deve ser feito para se alcançar ambientes com grau elevado de estabilidade.

3. Virtualização com Software Livre utilizando XEN

Neste capítulo são apresentadas as ferramentas para implantar virtualização a partir de softwares livre, mostrando as funcionalidades dessas ferramentas. É ainda apresentada a solução de alta disponibilidade para softwares livres, sistemas de controle e a dificuldade de implementação desta solução

4. Disponibilidade em ambientes virtualizados

Aqui é descrita e feita a implementação da solução, visando montar um ambiente de alta disponibilidade para virtualização, executando-se a análise de tolerância a falha do ambiente e descrevendo o comportamento deste ambiente.

5. Análise dos resultados obtidos

Após a obtenção dos resultados no capítulo anterior, é feita a análise da solução, ponderando os benefícios e apontando se os índices se apresentaram eficientes e tolerantes a falha, mantendo o ambiente estável. Nessa análise, leva-se em conta o Custo Total de Propriedade (TCO), a estabilidade, a disponibilidade e a dificuldade de implementação e manutenção do sistema virtualizado.

2 Virtualização e Alta disponibilidade

2.1 *Virtualização de sistemas operacionais*

Quando as empresas pensam em virtualização, estão buscando soluções que possibilitam uma redução do custo de seu parque computacional. Porém, não é somente a redução de custo que é propiciada com a virtualização dos sistemas operacionais, há também outros ganhos sensíveis e que devem ser levados em consideração, como a portabilidade, segurança, escalabilidade e principalmente a facilidade de gerenciamento das máquinas virtuais por estarem juntas, em um mesmo equipamento. [MARSHALL; RAYNOLDS, 2006]

Cada um dos sistemas que se utiliza da virtualização em seu funcionamento define o termo virtualização na forma que lhe convém, levando em consideração apenas interesses e particularidades que lhes compete. Mas, segundo o livro *Best Damn Server Virtualization Book Period*, virtualização é uma forma ou metodologia de dividir os recursos físicos de um computador em múltiplos ambientes de execução, aplicando uma ou mais tecnologias ou conceitos, como particionamento de hardware e software, compartilhamento de tempo de execução, simulação total ou parcial de máquina, emulação, qualidade de serviço, dentre outros. [RULE; DITTNE]

Ou seja, o objetivo da virtualização é compartilhar os recursos dos equipamentos.

2.2 *Objetivos da Virtualização*

Um dos grandes paradigmas existentes que deve ser levado em consideração para entender esse objetivo é a centralização ou não dos servidores

de uma empresa. No princípio da informática, toda capacidade de processamento das empresas se concentrava em seus *Main Frames*. Com o advento da microinformática, ocorreu uma migração, conhecida como “Down Sizing” onde os computadores de grande porte foram substituídos por máquinas de menor porte. Porém, viu-se que existem limitações para adoção de máquinas de pequeno porte, principalmente limitações físicas e de energia, tendo em vista que estas máquinas, mesmo em repouso, quando ligadas consomem uma quantidade considerável de energia, e que o tempo de inatividade dessas máquinas chega a ser de 85 a 90%, gerando um Custo Total de Propriedade (TCO) alto, e uma produção relativamente baixa. [WILLIAMS; GARCIA, 2007]

Desta forma, as empresas cada dia mais se utilizam da virtualização e buscam algumas funcionalidades em seu ambiente virtualizado, onde podemos obter:

- Portabilidade – A capacidade de migrar, de forma simples, de diferentes hardwares de vários fabricantes.
- Gerenciamento – A capacidade de gerenciar vários ambientes distintos em uma ferramenta única de hardware.
- Eficiência – O hardware hoje fica boa parte do tempo inativo, porém, com o uso da virtualização é possível majorar a eficiência do hardware de forma sensível, diminuindo o custo total de propriedade daquele equipamento virtualizado.[MARSHALL; RAYNOLDS, 2006]
- Confiabilidade – Definição de um ambiente de proteção entorno do aplicativo, criando instâncias de um aplicativo por servidor, isolando de forma

segura as falhas que por ventura ocorram. Outro ponto de confiabilidade é a criação de partições isoladas para cada aplicação.

- Segurança – A aplicação dos itens de confiabilidade gera um ambiente seguro, onde permite a segregação dos serviços, desatrelando o hardware do software e gerando assim um ambiente mais confiável para as aplicações em geral.

A Virtualização de infra-estruturas proporciona abstração entre os diferentes meios de computação como processamento, armazenamento de dados, rede e software. Ao implementar esse conceito, são eliminadas todas as restrições físicas e todos os meios ficam acessíveis como “agrupamentos lógicos”, que podem ser utilizados em qualquer hora e local. Além de outras vantagens, as soluções de Virtualização garantem maior capacidade de adaptação às constantes mudanças dos ambientes empresariais, mais eficiência e flexibilidade aos recursos de TI e redução dos custos associados.

2.3 Como funciona a virtualização

A virtualização de sistemas operacionais não é uma coisa que nasceu há pouco tempo. Desde a invenção do MULTICS, um dos primeiros sistemas operacionais desenvolvidos, uma concepção semelhante é mantida na forma de se particionar o hardware entre os sistemas virtualizados.

O particionamento ocorre, observando-se o nível mais baixo da operação, utilizando-se de micro-instruções específicas no processador, que possibilitam a criação de anéis de separação entre os níveis de execução das tarefas desenvolvidas para ocorrer a virtualização. Essa divisão em anéis,

desenvolvida inicialmente na implementação do MULTICS, possibilita a divisão do processador em 4 níveis distintos, chamados de anel0, anel1, anel2 e anel3 como é demonstrado na Figura1. Os anéis de 0 a 2, são conhecidos como Modo Supervisor e possuem acesso irrestrito às funções do processador, respeitando a hierarquia na execução, ou seja, uma tarefa rodando em anel0 não pode ser finalizada por uma tarefa rodando em anel1. O anel 3 é conhecido como Modo Usuário e possui acesso controlado às funções do processador. [MARSHALL; RAYNOLDS, 2006]

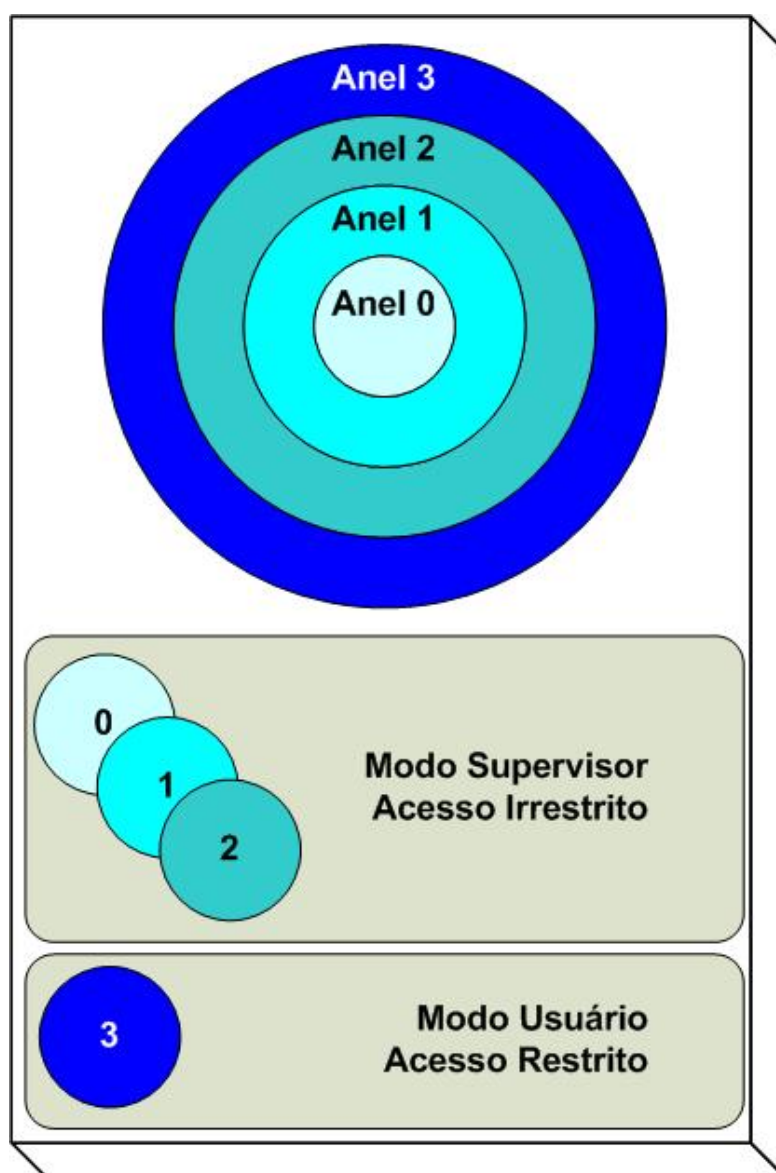


Figura 1- Particionamento baseado em anéis de privilégios, padrão do MULTICS. [MARSHALL; RAYNOLDS, 2006]

No processo de virtualização, a aplicação de controle das funções de virtualização também conhecida como VMM (*Virtual Machine Monitor*) é executada em Anel 0 possuindo assim acesso total às funções do processador, porém, às máquinas hospedadas neste servidor, são executadas em Anel 3, criando assim um ambiente de segregação dos processos executados e um nível a mais de estabilidade no sistema. A esse isolamento é dado o nome de ***Camada de Abstração***.

Existem duas formas distintas de criar esta camada de abstração entre o hardware e os sistemas operacionais instalados na máquina, na primeira forma, a VMM é instalada sobre um sistema operacional comum como é demonstrado na figura 2. Então os Sistemas Hóspedes instalados sobre a VMM passam a seguir a ordem de anéis estabelecida pelo sistema VMM, tendo em vista que o Anel 0 já é do sistema operacional onde a VMM foi instalada. Assim, se a VMM é instalada em Anel 3, esta emula mais uma série de anéis de 0 a 3 para hospedar seus sistemas virtualizados.

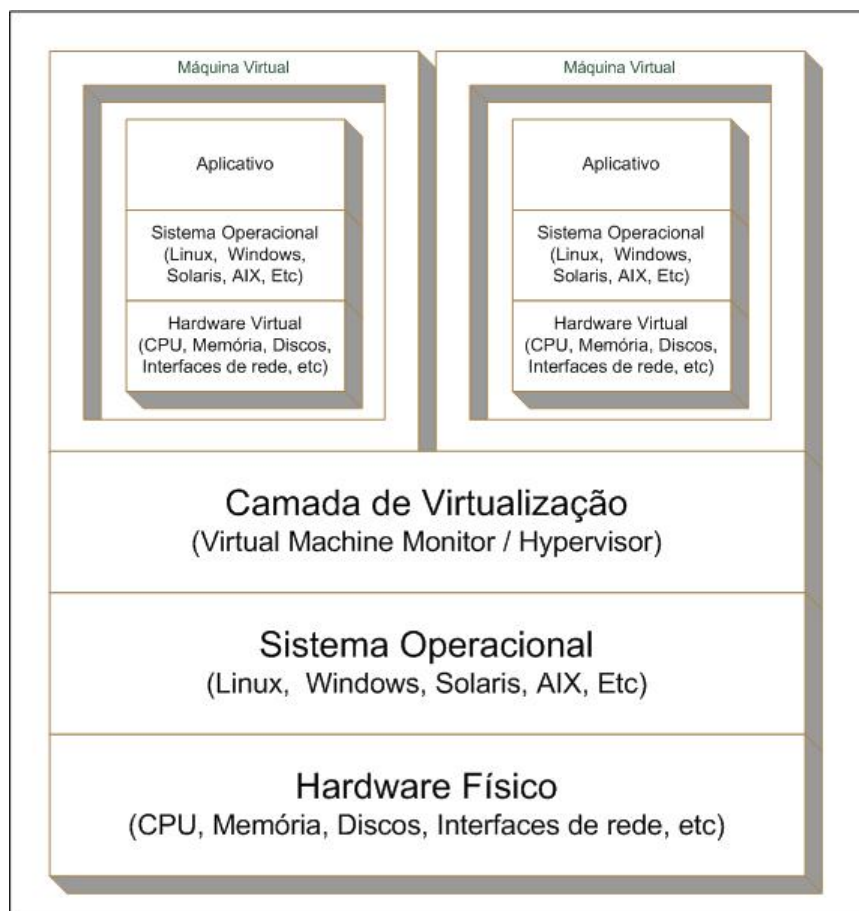


Figura 2 – Arquitetura de Servidor com virtualização baseada em VMM sobre Sistema Operacional [RULE; DITTNE]

Já em outra forma, conhecida como *Bare Metal*, conforme Figura 3, a VMM é instalada diretamente no hardware, sem utilizar outros sistemas operacionais, isso possibilita o total controle do hardware, sendo que a VMM vai realmente ser executada no Anel 0 e os sistemas virtualizados serão executados em Anel 3. Essa forma de implementação gera um ganho de desempenho, porém, softwares que possuem esse tipo de instalação costumam ser mais caros e de administração mais complexa, que os instalados em sistemas operacionais comuns.[RULE; DITTNE] e [MARSHALL; RAYNOLDS, 2006]



Figura 3 – Arquitetura de Servidor com virtualização instalada direta no Hardware [RULE; DITTNE]

2.4 Definido alta disponibilidade

A implementação de Alta Disponibilidade é um tema bastante estudado no ambiente computacional, não se tratando apenas da questão da estabilidade, mas também da disponibilidade e da confiabilidade do sistema. Este tema desperta grande interesse nas empresas e também no meio acadêmico.

Isso ocorre devido a Alta Disponibilidade ter uma ligação direta a crescente dependência dos computadores. Eles têm tido um papel crítico no mundo hoje, que vai desde um supermercado até os sistemas bancários. Com as inúmeras atividades que dependem de recursos computacionais, cresce cada vez mais o transtorno causado pela eventual falha dos sistemas.

A eficiência pode ser um grande diferencial. Um sistema que está sempre à disposição quando o cliente solicita, certamente é um diferencial. Infelizmente sistemas computacionais falham, mesmo os bem desenvolvidos. Um bom projeto prevê estas falhas e uma forma de contorná-las. É aí que se encaixa a Alta Disponibilidade.

A finalidade da alta disponibilidade é estar o maior tempo possível com o seu serviço disponível, sem deixar que seja interrompido. Deve-se saber que a alta disponibilidade não está relacionada apenas a um produto, mas sim a todo o sistema computacional. Existem diversas formas para melhorar a disponibilidade, porém, para que ocorra esse aumento é necessário um estudo sobre a configuração do sistema. [SZTOLTZ, 2008]

A disponibilidade de um sistema é o objetivo estratégico de diversas empresas, no que tange o desenvolvimento de soluções, podendo ser citadas as mais famosas, como IBM, VMware, Microsoft, CITRIX, RedHat. Essa disponibilidade não inclui somente o software, mas também o hardware, a rede, incluindo hubs, switches, etc. Ela pode ser classificada como alta disponibilidade básica, alta e contínua.

2.5 Formas de aferição de disponibilidade de sistemas

A Tabela 1 apresenta uma classificação de disponibilidade:

Tabela 1: Classificação de disponibilidade

Porcentagem	Tempo Desativado por ano	Classificação
99.5%	3.7 dias	Convencional
99.9%	8,8 horas	Disponível
99.99%	1 minutos	Disponibilidade Alta
99.999%	6 segundos	Resistente a Falhas
99.9999%	0.6 segundos	Tolerante a Falhas

A disponibilidade básica garante uma taxa de 99% a 99,9%. Qualquer sistema, que passe por uma série de testes e tenha como objetivo o mascaramento de falhas, sem mecanismos especiais, como softwares e hardwares, e esteja disponível no mercado garante essa disponibilidade. Já a alta disponibilidade possui uma taxa de 99,99% a 99,999%. Para se obter essa faixa de porcentagem é necessário utilizar mecanismos que detectam, recuperam e mascaram as falhas. A disponibilidade contínua possui uma taxa cada vez mais próxima 100%, porém, mesmo no ambiente ideal, a taxa de inoperância chega a ser tão pequena que, dependendo da necessidade do sistema, pode ser desprezível, mas nunca é de 100%, não existem sistemas infalíveis, todos os tipos de disponibilidade estão sujeitos à falhas. [PEREIRA, 2005]

Para aumentar a disponibilidade do sistema podem ser utilizados recursos de hardware e softwares específicos. [PEREIRA, 2005]

2.6 Parâmetros de medida de disponibilidade

Para se aferir a disponibilidade, é imprescindível a definição de parâmetros, onde possa ser feita a correlação entre os tempos de funcionamento e instabilidade do sistema. No mercado de tecnologia existem algumas definições estabelecidas, que são elas o MTBF e o MTTR.

O MTBF (*Mean Time Between Failures*) é o tempo médio entre falhas, e corresponde ao tempo médio entre as interrupções de funcionamento de equipamentos e sistemas. Deve ser medido entre a parada inicial de um sistema/aplicativo até próxima parada. Sistemas de alta disponibilidade costumam ter MTBF acima de 4000 horas, ou seja, 166,67 dias de funcionamento ininterrupto. A fórmula para medir o MTBF é:

$$MTBF = \sum (TT_n.abertura - TT_{n-1}.abertura) / (qtd\ TT - 1) \quad (1)$$

Onde TT significa Trouble Ticket, ou Abertura de Chamado, e a fórmula considera o intervalo de tempo entre uma ocorrência e outra de um defeito para um mesmo equipamento ou serviço.

Outra medida, o MTTR (*Mean Time To Repair*) mede o tempo médio de recuperação, e é estimado utilizando-se os prazos entre queda e recuperação de um sistema. Este parâmetro associado ao MTBF mede a eficiência do reparo em um componente ou serviço. Note que não basta reparar um componente ou serviço, se este apresenta constantes problemas de interrupção, é necessário que o componente ou serviço apresente baixa taxa de erro e rápida recuperação no seu funcionamento. Uma forma eficiente de se obter um MTTR bastante baixo é

proposta com a utilização de arquiteturas redundantes na concepção da solução. A

Fórmula para se medir o MTTR é:

$$\text{MTTR} = \sum \text{tempo de reparo} / \text{quantidade de defeitos} \quad (2)$$

Desta forma, a disponibilidade total do ambiente, passa a ser relativa à aferição a partir da verificação do tempo entre falhas e o tempo de solução do ambiente. A disponibilidade então pode ser determinada da seguinte forma:

$$\text{Disponibilidade} = \text{MTBF} / (\text{MTBF} + \text{MTTR}) \quad (3)$$

Veja que esse número sempre vai se aproximar de 1, tendo em vista que, em um ambiente de produção real, o tempo entre falhas tem que ser, sempre, muito maior que o tempo médio de recuperação, ou então o sistema não entra em produção.

Com estes dois índices, temos uma aferição numérica da disponibilidade do ambiente, porém, esta disponibilidade ainda é relativa, tendo em vista que se, neste momento, trata-se de um serviço de acesso exclusivamente durante o horário de expediente, se um sistema cai durante a noite, à indisponibilidade não chega a ser percebida. Esta Disponibilidade Relativa passa a depender de um fator externo, como uma chamada de cliente. Caso a indisponibilidade não seja percebida, pode ser desconsiderada na definição total de disponibilidade do ambiente. [FAGUNDES, 2004]

3 Virtualização com Software Livre Utilizando XEN

3.1 Sobre o XEN

O XEN é um monitor de máquinas virtuais (Virtual Machine Monitor), licenciado sobre os termos da GNU General Public Licence (GNU-GPL). Sua função é permitir a execução de vários sistemas operacionais (Linux, Solaris, Windows e BSD) ao mesmo tempo, dentro de outro sistema operacional, utilizando uma camada de abstração entre o software (SO Virtual) e os periféricos (hardware). O XEN surgiu dentro do projeto NetOS, projeto desenvolvido dentro da Universidade de Cambridge, que criou uma camada de abstração em que o sistema operacional funciona utilizando o servidor nas formas de um sistema distribuído. [Universidade de Cambridge. <http://www.cl.cam.ac.uk/research/srg/netos/>. Acessado em 01/06/2009]

Seu desenvolvimento é feito ativamente pelos membros da comunidade de software Livre, em conjunto com grandes empresas como Red Hat, IBM, AMD e HP. Hoje o XEN é propriedade da empresa CITRIX, porém não perdeu nenhuma funcionalidade nem seu uso ou acesso a código fonte foi restringido de qualquer forma. Seu mantenedor oficial antes da aquisição pela CITRIX era a Universidade de Cambridge.

A máquina raiz (principal) é denominada “Domínio”. Cada máquina virtual é denominada “DomínioU”. Ambas utilizam o kernel da máquina principal.

O conceito de Paravirtualização pode ser confundida com a emulação. Porém, a caracterização de paravirtualização como emulação é um erro, pois um sistema paravirtualizado limita-se a enviar instruções e um emulador, antes de

enviar instruções faz a interpretação e conversão dessas. Isso torna a virtualização mais eficiente e veloz. [Site do guia Livre do Governo Federal; 2007]

3.2 Formas de virtualização no XEN

O XEN possui a possibilidade de executar dois tipos de virtualização, a Paravirtualização e a Virtualização Completa (Full-Virtualization), cada uma com suas características, vantagens e desvantagens.

3.2.1 Virtualização Completa

A Virtualização completa, ou simplesmente a virtualização é o método mais conhecido de virtualização de sistemas operacionais, ele propicia um ambiente totalmente segregado do sistema hospedeiro, além de possuir maior facilidade e compatibilidade na instalação de uma maior quantidade de sistemas operacionais. Permite ainda maior portabilidade da máquina virtual entre ambientes distintos. Porém, o desempenho é diretamente comprometido, pois algumas chamadas diretas do sistema não possuem a possibilidade de serem virtualizadas, em função do nível de privilégio que o sistema atribui a elas, desta forma, há uma perda de desempenho, pois os sistemas emulam os privilégios de instrução x86. Na utilização deste tipo de virtualização, com software livre, é necessária a instalação de processadores preparados para isso, com instruções específicas para a virtualização.

3.2.2 Paravirtualização

A Paravirtualização promove uma virtualização parcial entre o sistema virtualizado e a camada de hardware, o principal diferencial é a definição do espaço virtualizado, isso garante que a máquina virtualizada possui um espaço único para ela. A paravirtualização também garante grande velocidade à máquina virtualizada no acesso as interfaces de E/S do sistema. Porém, na implementação das máquinas virtuais, é necessária alterações no kernel do sistema operacional das máquinas com sistemas virtualizados para que estas funcionem, além de possuir baixa portabilidade [WILLIAMS; GARCIA, 2007]

Ao utilizar a Paravirtualização no XEN é necessário instalar um Kernel com instruções já definidas para que a Paravirtualização ocorra. Existem diferenças na instalação do Kernel X86 quando esse é para 32 bits ou quando é para 64 bits. Com Full Virtualization, nas mesmas arquiteturas de hardware que suportem virtualização assistida, não é preciso modificar o sistema. Existem também versões do XEN para outros sistemas como FreeBSD, NetBSD ou SUN/Solaris.

3.3 Ferramentas de migração de servidores em tempo de execução

– Máquinas Vivas

Em ambientes virtualizados, é possível implementar uma funcionalidade que simplifica a administração do ambiente computacional físico. A opção de utilização de migração de servidores em tempo de execução, ou máquinas vivas, permite ao administrador de sistemas migrar todas as máquinas virtualizadas de uma máquina física para outra, tendo assim a possibilidade de

desligar uma máquina física sem ter a indisponibilidade dos serviços que são prestados.

A migração de máquinas vivas ocorre em estágios, como demonstrado na Figura 4, onde estes devem acontecer em um espaço de tempo entre 50 e 300 ms para que em nenhuma requisição TCP das aplicação ocorra *timeout*. [WILLIANS; GARCIA, 2007]

Os estágios são sequenciais e ocorrem da seguinte maneira:

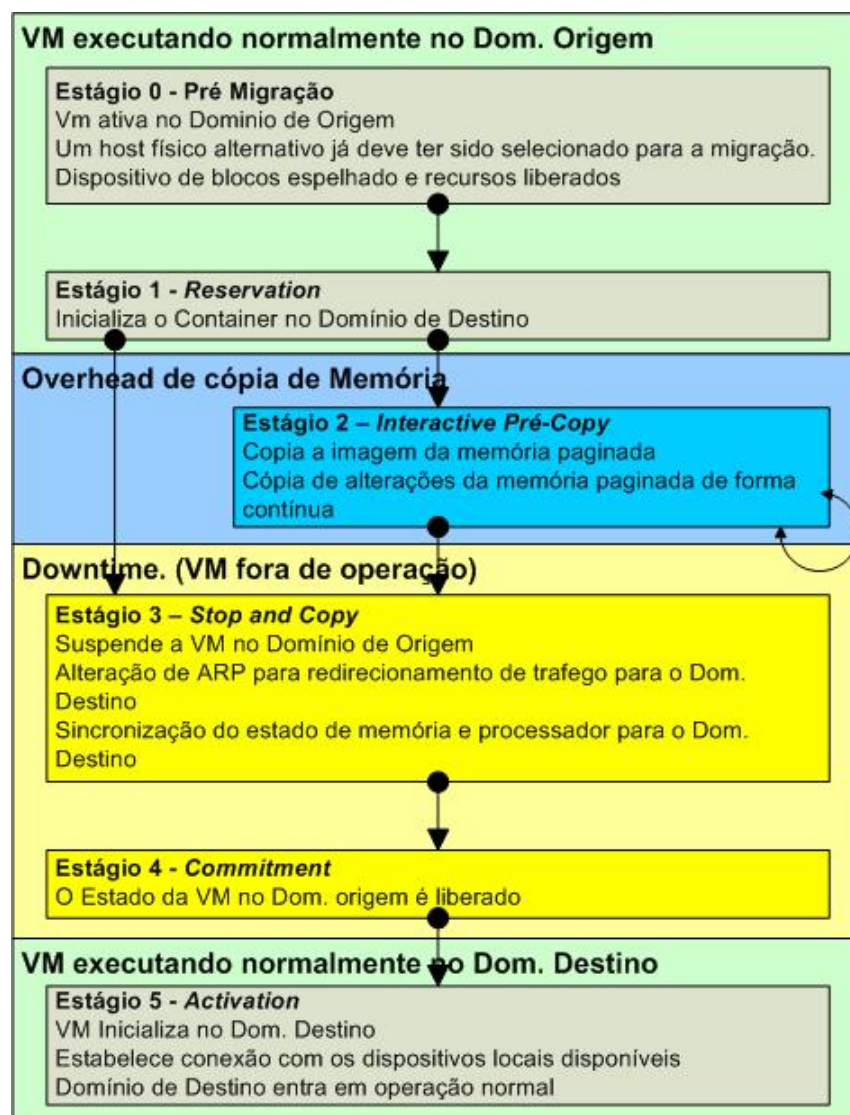


Figura 4 – Estágios do processo de Migração de Máquinas Virtuais [WILLIANS; GARCIA, 2007]

Estágio 1 – *Reservation* – O domínio de origem solicita ao domínio de destino verificar se possui recursos disponíveis, então o host de destino cria uma VM com os recursos necessários a criação da máquina virtual. Caso haja qualquer falha nessa fase, a máquina virtual no domínio de origem continua funcionando sem nenhum prejuízo ao funcionamento da máquina virtual

Estágio 2 – *Iterative Pre-Copy* – Se o domínio de destino possui recursos suficientes, é iniciada a fase de cópia de páginas de memória do domínio de origem para o domínio de destino.

Estágio 3 – *Stop-and-Copy* – Após a cópia das páginas de memória, somente são copiadas as alterações nessas páginas para manter a igualdade nas máquinas virtuais. São copiadas também todos os estados do processador do domínio de origem para o domínio de destino. O domínio de origem continua sendo considerado o domínio ativo, e caso seja necessária a desativação da migração, nada ocorre. Nesta fase as conexões TCP são redirecionadas para o domínio de destino.

Estágio 4 – *Commitment* – O domínio de destino informa ao domínio de origem que recebeu a imagem da máquina virtual de forma consistente. O domínio de origem entende esta informação como a conclusão do processo de migração, descartando a VM do domínio de origem e ativando a VM do domínio de destino.

Estágio 5 – *Activation* – O domínio de destino agora se torna o principal e roda os scripts para ativação dos *device drives* e publicação da migração do IP virtualizado.

3.4 Ferramentas de alta disponibilidade

Para a criação de uma ambiente de alta disponibilidade no XEN é necessária a utilização de sistemas desenvolvidos com esta finalidade, que são:

3.4.1 OCFS2 - sistema de arquivo compartilhado

Oracle Cluster File System 2 ou OCFS2 é o sistema de arquivos, inicialmente desenvolvido pela empresa Oracle, utilizado largamente na implantação de ambiente de banco de dados *Oracle RAC (Real Application Cluster)*, e que hoje é mantido por ela e por uma comunidade de desenvolvedores, para operar de forma compartilhada entre servidores, funcionando em qualquer opção de sistema operacional linux, propiciando a interoperabilidade de sistema de cluster.

Este sistema de arquivos nasceu da evolução do OCFS, que inicialmente funcionava apenas para criação de ambiente Oracle RAC, para funcionar em ambiente de disco compartilhado, definindo uma *Storage Área Networking (SAN)*. Por sua natureza é um sistema de arquivo compartilhado que diferentemente dos sistemas de arquivos locais, são preparados para funcionar de forma distribuída, sendo acessados por vários servidores, propiciando alta disponibilidade.

O sistemas de arquivo em cluster OCFS funciona de forma simétrica, ou seja, permitindo a todos os nós que compõem o cluster fazer acesso de escrita e leitura no sistema de arquivos.

Não possui a funcionalidade de escrita de dados paralela, que permitiria que duas aplicações escrevessem juntas em um mesmo arquivo de

destino, mas possibilita, através do controle de *lock*, a manutenção da consistência dos dados gravados e a implementação de *faill-over* para utilização em aplicações, como por exemplo, Servidor de Email, Servidor de Máquinas Virtuais e Servidor de Arquivos. [MASHRAN, 2009]

A escolha deste sistema de arquivos considerou as seguintes funcionalidades:

- Journaling – funcionalidade de fazer *writeback* em dados do sistema operacional. Isso garante que, em caso de falha da aplicação ou falta de energia os dados não serão corrompidos
- Tamanho do bloco variável e *inodes* gigantes – Capacidade de implementar *inodes* de tamanho variável, variando de 512 bytes a 4Kbytes
- Arquitetura Neutra – Possibilidade de implantação em várias distribuições Linux rodando sobre qualquer tipo de hardware.
- Administração completa e simplificada.

3.4.2 Heart Beat2 – Monitoramento do cluster

O aplicativo *Heartbeat*, que é uma aplicação responsável por monitorar um cluster de máquinas, verificando quais delas estão acessíveis através de pacotes *keepalive udp*, enviados de tempos em tempos, (chamados *heartbeats*). Quando a máquina principal de um cluster fica inacessível, o *heartbeat* rodando

em alguma das outras máquinas do cluster será responsável por ativar serviços e configurações nesta para que ela assuma o papel da primeira.

Um IP compartilhado (virtual) é associado à interface da máquina primária enquanto ela estiver ativa. No momento em que a máquina secundária perceber que a primária está inacessível, ela adquire o IP virtual da máquina primária e passa a responder pelo serviço. Quando a máquina primária voltar à atividade, existem duas possibilidades: ela assume o IP virtual automaticamente ou espera por uma intervenção humana para tal. Essa configuração será abordada mais a frente. [The High Availability Linux Project, <http://www.linux-ha.org/>. Ultimo Acesso em 18/04/2008]

3.4.3 iSCSI – Internet Small Computer System Interface

O protocolo de transmissão de blocos, iSCSI, acrônimo que significa Internet Small Computer System Interface, é um protocolo de mapeamento do Procedimento Remoto de Invocação SCSI sobre o protocolo TCP. Ou seja, comandos SCSI são transportados pelas requisições iSCSI e blocos de respostas e status são transportados pelas respostas iSCSI. [SATRAN, METH, SAPUNTIZAKIS, CHADALAPAKA, ZEIDNER; 2004]

O protocolo iSCSI, como baseia-se no procedimento padronizado do SCSI, também é um protocolo cliente-servidor, ou seja, necessita de uma conexão com uma aplicação servidora que, a partir dela os dados são transportados para o cliente. Desta forma, o protocolo iSCSI se comunica com o servidor, através de uma ou mais conexões TCP estabelecidas, e transporta os comandos SCSI. Os datagramas transportados são encapsulados por um cabeçalho TCP/IP e é

transportado pela rede, até o destino iSCSI. O iniciador iSCSI abre o pacote TCP, interpreta o cabeçalho do pacote e verifica que ali existe um bloco SCSI. [SATRAN, METH, SAPUNTIZAKIS, CHADALAPAKA, ZEIDNER; 2004]

Desta forma, o protocolo iSCSI é considerado um protocolo mais lento que os protocolos de dados padrão, como por exemplo o protocolo FC, porém, dado seu simples procedimento de implantação e sua possibilidade de funcionamento sem a necessidade de aquisição de equipamentos especializados, o mercado para o protocolo iSCSI tem crescido a cada dia que empresas optam pela utilização desta tecnologia. [Site do guia Livre do Governo Federal; 2007]

Novas tecnologias estão surgindo para tornar mais funcional o protocolo iSCSI. A tecnologia de TOE (TCP Offload Engine), desenvolvida pela Intel para agilizar as conexões TCP, fazendo a interpretação do cabeçalho do protocolo TCP/IP já através do processador da placa de Rede, redes de 10Gb Ethernet e drives mais eficientes, facilitam cada dia mais a implantação e utilização dessa tecnologia em empresas de todos os tamanhos. [SATRAN, METH, SAPUNTIZAKIS, CHADALAPAKA, ZEIDNER; 2004]

4 Instalando o Ambiente Virtualizado.

4.1 Topologia da Solução

Na busca de um ambiente estável para as máquinas virtuais funcionarem com alta disponibilidade, como descrito anteriormente, optou-se por uma solução baseada em softwares livres, sem a necessidade de dispor de recursos financeiros, para montar toda a infra-estrutura de software.

Dentro dessa infra-estrutura, optou-se também por *hardwares* de baixo custo, desenvolvido inicialmente para o ambiente residencial, mas que é capaz de servir facilmente a um ambiente corporativo sem nenhum tipo de customização ou melhoria a partir de sua compra original.

A topologia adotada para o ambiente também é uma topologia simples, sem grandes customizações ou particularidades que de qualquer forma inviabilize a qualquer empresa, de qualquer porte, a instalação deste ambiente.

4.1.1 Topologia Física

A topologia física adotada está demonstrada na Figura 5 abaixo.

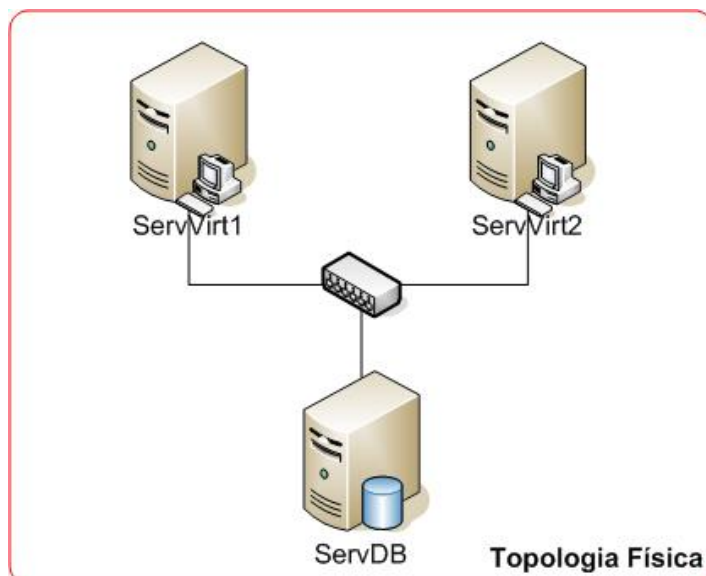


Figura 5 – Topologia Física adotada entre os servidores

O cabeamento adotado é UTP categoria 5, funcionando em rede com topologia Estrela, a 100Mbps, no padrão 100BASE-TX. O concentrador é um Switch Wireless marca Linksys modelo WRT54G como mostrado na figura 6. Neste equipamento não houve a necessidade de configuração específica para seu funcionamento neste ambiente.



Figura 6 – Switch Linksys WRT54G utilizado no projeto

Os equipamentos utilizados como servidores de máquina virtual e Banco de Dados, são de dois modelos. Na figura 7 está a imagem de dois dos

servidores adquiridos, marca Intelbras, adquiridos no supermercado Extra Park, como pode ser verificado no anexo II, com a configuração descrita na Tabela 2. Isso mostra que não é necessário o dispêndio de recursos financeiros volumosos, para se implementar uma solução de Cluster de Alta Disponibilidade para máquinas virtuais. O outro computador adotado, na instalação deste cluster, é um notebook, da marca Sony, modelo VAIO VGN-CR320E.

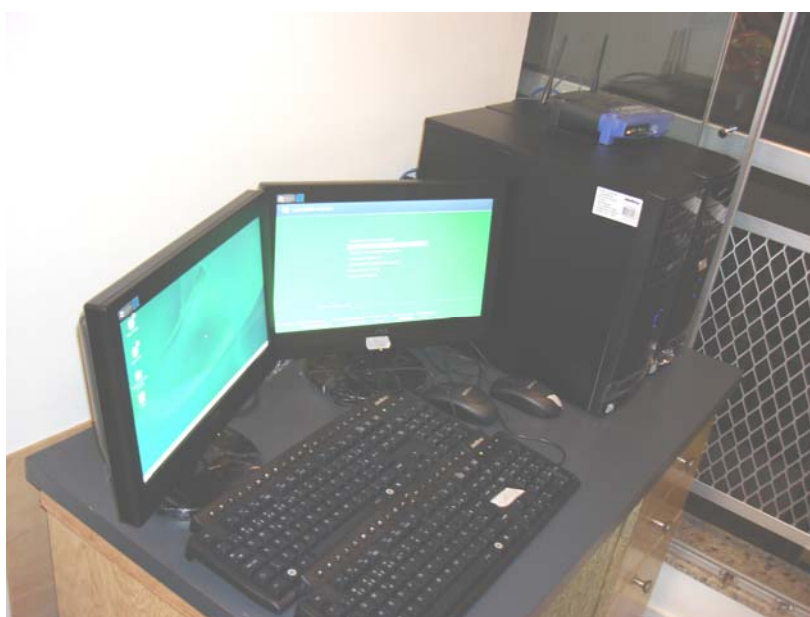


Figura 7 - Servidor de Virtualização e Banco de dados

Tabela 2 – Configuração dos equipamentos que compõem o cluster

Configuração	ServVirt1	ServVirt2	ServDB
Fabricante	Intelbras	Intelbras	Sony
Processador	Intel Celeron 450	Intel Celeron 450	Intel Core 2 DUO T7250
Clock	2,20Ghz	2,20Ghz	2,0 Ghz
Barramento	800 Mhz	800 Mhz	800 Mh
Cache	512 Kb	512 Kb	2 MB
Memória RAM	2GB DDR 2	2GB DDR 2	2 GB DDR 2
Disco rígido	160 GB	160 GB	160 GB

4.1.2 Topologia Lógica

Na topologia lógica adotada funciona, tendo como base do funcionamento os protocolos TCP e UDP, nas funcionalidades de iSCSI e HeartBeat como demonstrada na figura 8.

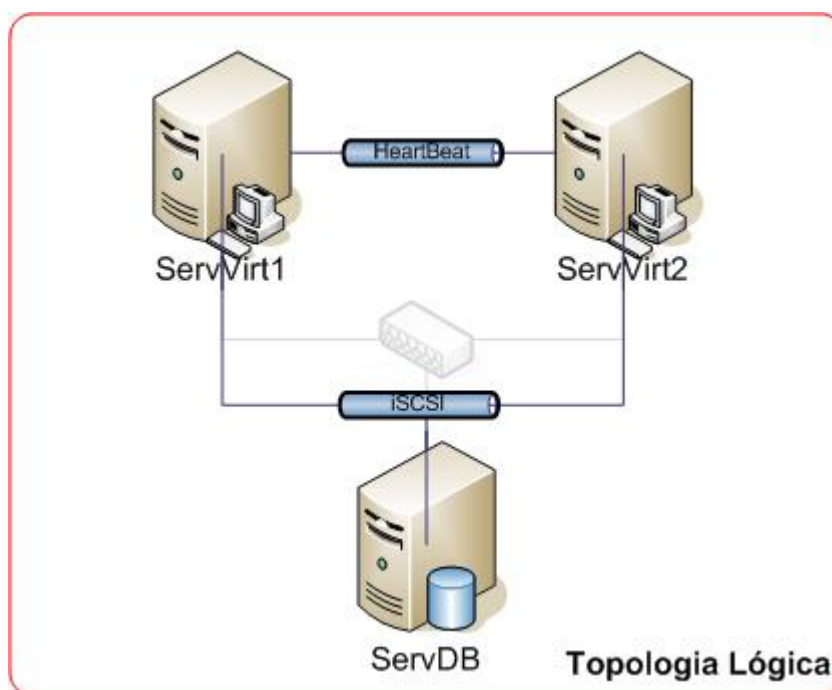


Figura 8 – Topologia lógica da solução apresentada.

Na funcionalidade iSCSI, o servidor de Imagens Virtuais ServDB, possui instalado nele o Destino iSCSI, ou seja, as conexões são estabelecidas com ele e, a partir dele, são montados os discos virtuais nos servidores ServVirt1 e ServVirt2 através dos Iniciadores iSCSI. Desta forma, os servidores ServVirt1 e ServVirt2 acessam os dados do servidor ServDB como se estivessem em um disco local.

Para se manter a boa integridade das informações desses discos virtuais, o sistema de arquivos é o OCFS2, que controla o acesso de cada uma das máquinas, não permitindo que esses dados sejam corrompidos com acessos

simultâneos indiscriminados, fazendo o controle dos locks e journaling dos dados gravados.

Entre os servidores ServVirt1 e SevVirt2, a aplicação HeartBeat verifica a estabilidade do sistema fazendo algumas análises, dentre elas, as seguintes:

- Abre uma conexão na porta da aplicação para ver se a aplicação definida para aquele sistema está funcionando
- Envia um pacote UDP informando que a máquina está funcionando.

Se a máquina está funcionando, mas a aplicação não responde, pode ser dado comando para reiniciar a aplicação, se este comando não surtir efeito, por exemplo, na máquina ServVirt1, a máquina ServVirt2 manda um comando para que o sistema seja realmente derrubado e, a partir daí, a aplicação começa a ser executada no ServVirt2.

Ou seja, caso a aplicação de máquinas virtuais pare de funcionar, por exemplo, no equipamento ServVirt1 as máquinas virtuais que estavam em execução neste equipamento, passarão a funcionar na máquina ServVirt2, seguindo os passos da migração de máquinas vivas citada no capítulo 4.3.

4.2 Instalação do Sistema Operacional para Ambientes Virtualizados

Neste enfoque, então, foi selecionado o Sistema Operacional GNU/Linux Novell Open Suse, na versão 11.1. Este sistema operacional possui algumas vantagens, como a existência de uma empresa desenvolvendo soluções

para ele, o que torna o sistema operacional bem estável e com atualizações rápidas e efetivas para falhas encontradas. Outra vantagem é a documentação existente para este sistema operacional concentrada no web-site da Novell.

A mesma versão do sistema operacional foi instalada nas três máquinas que compõem o cluster. A seleção dos pacotes a serem instalados, além dos pacotes padrão do sistema operacional, foram os seguintes:

- Para instalar o sistema de virtualização:

```
xen-doc-html-3.3.1_18494_03-1.8
kernel-xen-extra-2.6.27.7-9.1
xen-3.3.1_18494_03-1.8
xen-tools-3.3.1_18494_03-1.8
kernel-xen-base-2.6.27.7-9.1
xen-libs-3.3.1_18494_03-1.8
kernel-xen-2.6.27.7-9.1
xen-doc-pdf-3.3.1_18494_03-1.8
```

- Para instalar o Sistema de Arquivos OCFS2

```
ocfs2-tools-1.4.1-6.10
ocfs2console-1.4.1-6.10
```

- Para instalar o Heart Beat

```
heartbeat-common-2.99.3-1.6
heartbeat-2.99.3-1.6
heartbeat-resources-2.99.3-1.6
libheartbeat2-2.99.3-1.6
```

- Para instalar o iSCSI

```
yast2-iscsi-client-2.17.5-1.2
iscsitarget-kmp-xen-0.4.15_2.6.27.7_9.1-89.10
iscsitarget-0.4.15-89.10
open-iscsi-2.0.870-21.1
iscsitarget-kmp-pae-0.4.15_2.6.27.7_9.1-89.10
yast2-iscsi-server-2.17.3-1.25
```

4.3 Configuração do ambiente para funcionamento

4.3.1 Configurando o Sistema Operacional

Tendo em vista que o sistema operacional desses equipamentos vai trabalhar em cluster, é necessário fazer algumas sincronizações para que os sistemas tenham horários e alguns dados iguais. Na configuração do horário, para

que os servidores fiquem com os mesmos horários foi feita a configuração do cliente de NTP da máquina sincronizando com o observatório nacional.

4.3.2 Configuração do NTP

A configuração do cliente NTP é fundamental para a manutenção da estabilidade do ambiente. Como todos os servidores vão fazer escrita simultânea no mesmo disco, é fundamental que estes servidores que vão escrever em momentos simultâneos no disco, possuam uma referência de relógio idêntica.

Segue abaixo uma cópia do arquivo ntp.conf com a configuração dos clientes de NTP.

```
#####
## /etc/ntp.conf
##
## Sample NTP configuration file.
## See package 'ntp-doc' for documentation, Mini-HOWTO and FAQ.
## Copyright (c) 1998 S.u.S.E. GmbH Fuerth, Germany.
##
## Author: Michael Andres, <ma@suse.de>
##         Michael Skibbe, <mskibbe@suse.de>
##
#####
server a.ntp.br iburst
server b.ntp.br iburst
server c.ntp.br iburst

driftfile /var/lib/ntp/drift/ntp.drift # path for drift file

logfile    /var/log/ntp          # alternate log file
keys /etc/ntp.keys              # path for keys file
trustedkey 1                    # define trusted keys
requestkey 1                    # key (7) for accessing server
variables
```

4.3.3 Configuração do arquivo hosts

Todos os servidores devem se conhecer, no que tange o seu nome de acesso e endereço IP. Como não foi definido entre os servidores um servidor de

DNS primário, a opção pela definição dos endereços no arquivo “hosts” faz com que os equipamentos compartilhem deste conhecimento mútuo.

Segue abaixo uma cópia do arquivo “hosts”, que é muito similar entre os servidores, exemplificando as configurações que foram feitas.

```
#
# hosts          This file describes a number of hostname-to-address
#                mappings for the TCP/IP subsystem.  It is mostly
#                used at boot time, when no name servers are running.
#                On small systems, this file can be used instead of a
#                "named" name server.
# Syntax:
#
# IP-Address    Full-Qualified-Hostname  Short-Hostname
#
127.0.0.1      localhost
# special IPv6 addresses
::1           localhost ipv6-localhost ipv6-loopback
fe00::0       ipv6-localnet
ff00::0       ipv6-mcastprefix
ff02::1       ipv6-allnodes
ff02::2       ipv6-allrouters
ff02::3       ipv6-allhosts
127.0.0.2     linux-sqng.site linux-sqng
192.168.25.11 ServVirt1
192.168.25.12 ServVirt2
```

4.3.4 Configuração do Particionamento do disco

O particionamento do disco rígido do servidor deve ocorrer de tal forma que haja, na instalação, uma partição específica e destacada para a instalação, sobre ela, das máquinas virtuais. Essa partição não pode ser criada e definida durante a instalação, no caso da criação de máquinas virtuais em um cluster de alta disponibilidade, tendo em vista que deve funcionar com sistema de arquivo OCFS2, sistema este que não é padrão dentre os sistemas de arquivos do Linux.

Desta forma, após a instalação, executa a funcionalidade do Yast chamada “Particionador”, e faz a definição do disco que se deseja criar, como demonstrado na Figura 9, a partir da seguinte sequência de comandos:

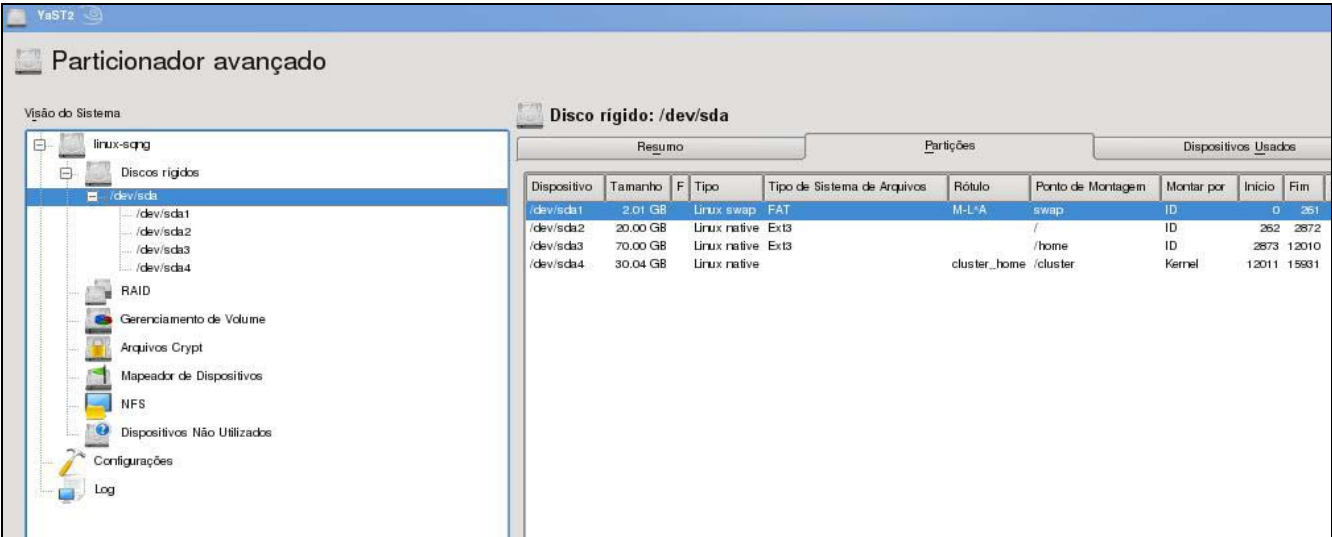


Figura 9 – Tela de configuração do Particionador

Tabela 3 – Procedimento de configuração do particionador.

No Servidor que vai hospedar as imagens, cria-se uma partição de disco para montar o ambiente
Entre no particionador
Selecione o disco rígido com espaço direcionado para se hospedar as imagens. No caso dos servidores adotados, /dev/das
Clique no botão “adicionar...”
Defina o espaço do disco
Informe o Tipo de Sistema de arquivos como sendo “Linux”
Aplique a configurações e saia do configurador.

Este disco não deve estar formatado e nem ter atribuído a ele, nenhum ponto de montagem.

Após a definição de um espaço de disco, que no caso do servidor ServDB é o volume /dev/sda4, deve-se formatar a partição com o sistema de arquivo OCFS2. Para tanto, é executado o seguinte comando

```
mkfs.ocfs2 --fs-feature-level=max-compat -b 4K -C 32K -N 2 -L cluster_home /dev/sda4
```

Onde cada um dos atributos do comando significa;

“fs-feature-level=Max-compatible” habilita a compatibilidade da partição com todas as versões do OCFS2, com isso se reduz a chance de acontecerem erros de funcionamento da aplicação por qualquer incompatibilidade de versão.

-b (block-size) ou tamanho do bloco de dados que será guardado no disco. O bloco é a um tamanho definido para uma sequência de bytes do sistema operacional, fazendo uma abstração da forma real em que o dado é guardado no disco. Não se recomenda a opção por blocos muito pequenos, pois esses geram uma fragmentação muito grande dos dados no disco, prejudicando o armazenamento de dados.

-C (Cluster-Size) O Cluster que representa a menor quantidade de espaço em disco que pode ser alocada para armazenar um arquivo, ou seja, no armazenamento de dados, um arquivo não utiliza apenas um setor de disco, mas uma sequência de setores, como mostrado na figura 10 pela legenda D. Quanto menor, em bits, o tamanho do cluster, mais eficiente será o armazenamento, no disco, das informações. Porém, quando se deseja armazenar arquivos muito grandes, como é o caso de imagens de Sistema Operacionais ou Bancos de Dados, se recomenda o uso dos clusters do disco maiores.

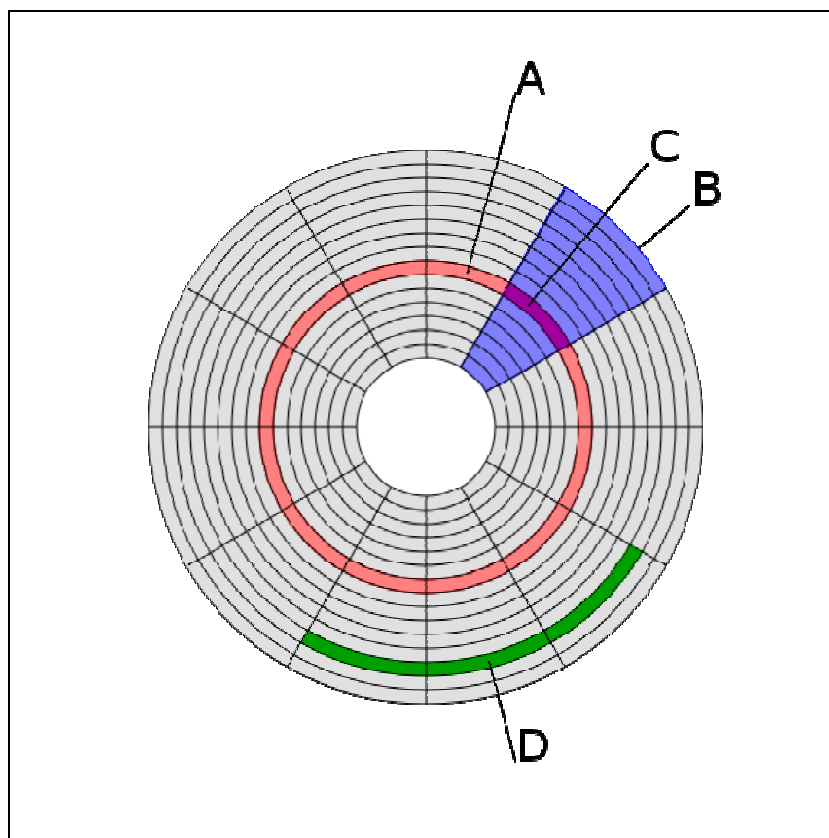


Figura 10 – Estrutura de divisão do Disco Rígido.

-N (number of node slots) Define o número máximo de nós que irão fazer a montagem do sistema de arquivos e poderão assim fazer acesso simultâneo no disco montado. O padrão de montagem do sistema de arquivos é 4.

-L (volume lable) Essa informação define o nome que a partição passará a ter após a montagem do disco. Não é obrigatória, mas facilita a localização da informação no servidor. No caso do cluster montado, definiu-se o nome `cluster_home`.

`/dev/sda4` é a partição física do disco na qual é feita a partição com o sistema de arquivos `ocfs2`.

4.3.5 Configurando o Cluster

Para o perfeito funcionamento deste ambiente, é necessária a criação e configuração de um cluster. Neste procedimento foi utilizado o sistema de arquivos OCFS2 para compartilhar o conteúdo entre os servidores e ainda o protocolo de transmissão de dados em blocos iSCSI.

4.3.5.1 Configuração do OCFS2

Para a configuração do cluster, é fundamental que já se tenha instalado no servidor o aplicativo ocfs2-tools.

Na criação do cluster é necessário seguir os seguintes passos:

- Executar o binário `/etc/init.d/o2cb` configure e responder as seguintes perguntas:
 - o `Load O2CB driver on boot (y/n) [n]: y`
 - o `Cluster to start on boot (Enter "none" to clear) []: ocfs2`
- A resposta deve ser:
 - o `Writing O2CB configuration: OK`

Após isso, para colocar o cluster em modo online, dá-se o seguinte comando:

- o `/etc/init.d/o2cb online ocfs2`
- A resposta deve ser
 - o `Starting cluster ocfs2: OK`

Feito isso, o sistema de cluster entrará em modo online e passa a funcionar, mas não está ainda definido quais são os nós desse cluster. Então é feita a edição do arquivo `/etc/ocfs2/cluster.conf`. Neste arquivo, é definido quem são os nós do cluster que está sendo criado.

Abaixo, é apresentado o arquivo de configuração do cluster gerado para o projeto. Todos os arquivos de configurações das máquinas são idênticos e compartilhados a partir da máquina ServDB.

```
node:
  ip_port = 7777
  ip_address = 192.168.25.10
  number = 0
  name = ServVirt1
  cluster = ocfs2

node:
  ip_port = 7777
  ip_address = 192.168.25.11
  number = 1
  name = ServVirt2
  cluster = ocfs2

node:
  ip_port = 7777
  ip_address = 192.168.25.12
  number = 2
  name = ServDB
  cluster = ocfs2

cluster:
  node_count = 3
  name = ocfs2
```

4.3.5.2 Configuração do iSCSI

4.3.5.2.1 Configuração do Destino iSCSI

No Servidor de Imagens, para se fazer a comunicação e a transação dos arquivos como blocos de dados, faz-se a instalação do Destino iSCSI. Esse protocolo, iniciado pelo aplicativo de Destinos iSCSI, é o padrão de comunicação de dados pelo qual os servidores de máquinas virtuais, neste projeto ServVirt1 e ServVirt2, irão se comunicar com o servidor de imagens, chamado ServDB.

A instalação desse aplicativo é feita pelo pacote “iscsitarget” e sua configuração é bem simples.

Executa-se o aplicativo de Destino iSCSI no Yast, que possui a interface mostrada na figura 11. Altera-se a inicialização deste protocolo para ocorrer na inicialização do computador e na Aba “Destinos”, é editada a LUN0.

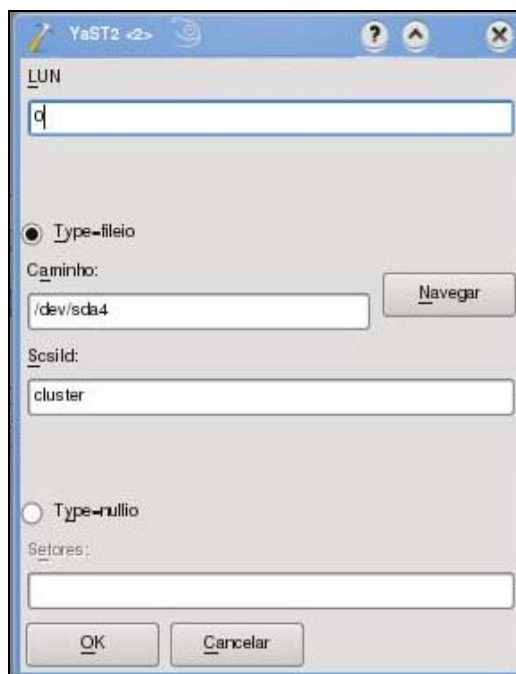


Figura 11 – Destino iSCSI através do configurador YAST

Ao editar o destino LUN0, deve-se definir o caminho físico que aquela LUN corresponde, ou seja, a partição de disco, que no caso deste projeto é /dev/sda4.

4.3.5.3 Configuração do Iniciador iSCSI

Após a configuração do Destino iSCSI, nos servidores de máquina virtual, o cliente iSCSI deve ser configurado. A configuração do cliente se dá da seguinte maneira:

Executa-se o aplicativo de Iniciador iSCSI no Yast, que possui a interface mostrada na figura 12. Acessada a aba Destinos Conectados e pressiona-se o botão adicionar. Adicione o IP do destino. A porta padrão é a 3260. Após

completar este passo, é necessário definir em qual momento será estabelecida a conexão, que, no caso deste projeto, é definido ao iniciar.

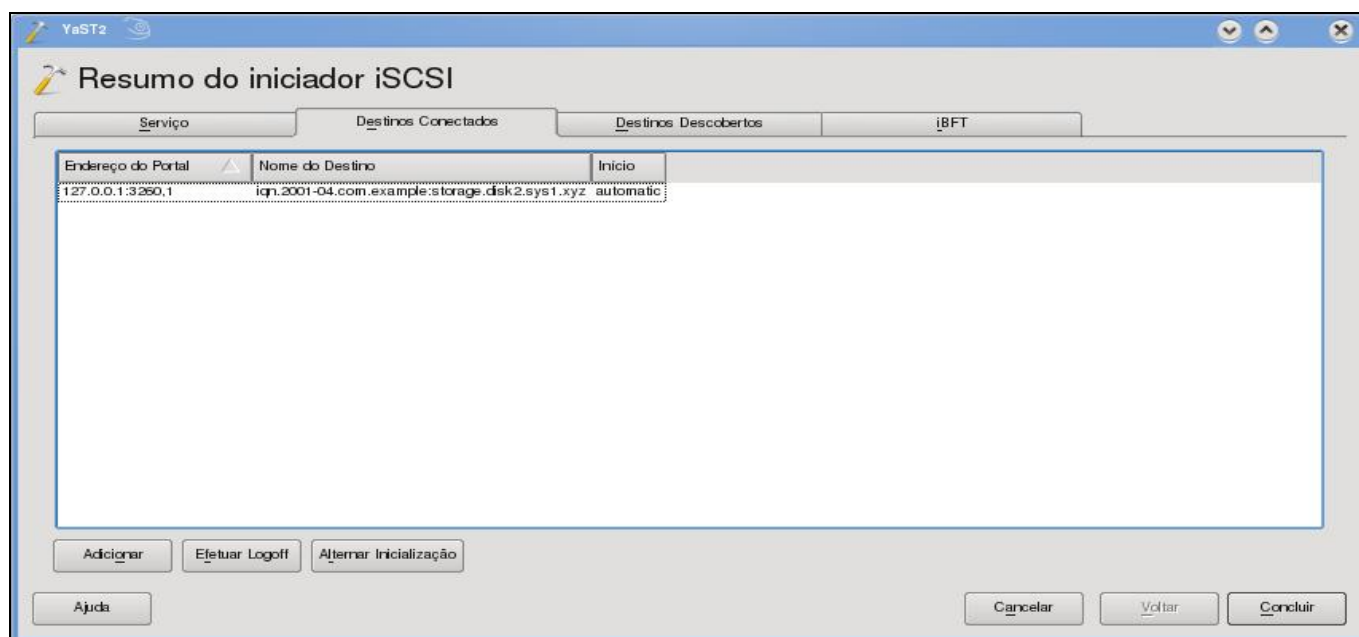


Figura 12 – Iniciador iSCSI através do configurador YAST

Após esta configuração ser executada, o aplicativo escreve um arquivo de configuração com todos os dados necessários para a inicialização do Iniciador iSCSI. Segue abaixo a configuração deste arquivo para um dos servidores estabelecidos no cluster.

```
node.name = iqn.2001-04.com.example:storage.disk2.sys1.xyz
node.tpgt = 1
node.startup = automatic
iface.hwaddress = default
iface.ipaddress = default
iface.iscsi_ifacename = default
iface.net_ifacename = default
iface.transport_name = tcp
node.discovery_address = 192.168.25.10
node.discovery_port = 3260
node.discovery_type = send_targets
node.session.initial_cmds_n = 0
node.session.initial_login_retry_max = 8
node.session.cmds_max = 128
```

```

node.session.queue_depth = 32
node.session.auth.authmethod = None
node.session.timeo.replacement_timeout = 120
node.session.err_timeo.abort_timeout = 15
node.session.err_timeo.lu_reset_timeout = 20
node.session.err_timeo.host_reset_timeout = 60
node.session.iscsi.FastAbort = Yes
node.session.iscsi.InitialR2T = No
node.session.iscsi.ImmediateData = Yes
node.session.iscsi.FirstBurstLength = 262144
node.session.iscsi.MaxBurstLength = 16776192
node.session.iscsi.DefaultTime2Retain = 0
node.session.iscsi.DefaultTime2Wait = 2
node.session.iscsi.MaxConnections = 1
node.session.iscsi.MaxOutstandingR2T = 1
node.session.iscsi.ERL = 0
node.conn[0].address = 192.168.25.10
node.conn[0].port = 3260
node.conn[0].startup = onboot
node.conn[0].tcp.window_size = 524288
node.conn[0].tcp.type_of_service = 0
node.conn[0].timeo.logout_timeout = 15
node.conn[0].timeo.login_timeout = 15
node.conn[0].timeo.auth_timeout = 45
node.conn[0].timeo.noop_out_interval = 5
node.conn[0].timeo.noop_out_timeout = 5
node.conn[0].iscsi.MaxRecvDataSegmentLength = 131072
node.conn[0].iscsi.HeaderDigest = None
node.conn[0].iscsi.DataDigest = None
node.conn[0].iscsi.IFMarker = No
node.conn[0].iscsi.OFMarker = No

```

Após essa configuração estabelecida nos servidores, o ponto de montagem é definido nos Servidores de Máquinas virtuais, chamados de ServVirt1 e ServVirt2, e é neste ponto de montagem onde serão instaladas e salvas as máquinas virtuais que farão parte do cluster.

A montagem do cluster se dá com o comando `mount.ocfs2 /dev/sdc /cluster` onde

`/dev/sdc` é o disco virtual que é criado no servidor quando está acessando o disco via iSCSI

`/cluster` é o ponto de montagem

4.3.6 Configuração do Sistema de Migração de Máquinas Vivas

A configuração do XEN para permitir a migração de máquinas vivas é feito a partir da configuração do arquivo `xend-config.sxp`. Neste procedimento, são definidos os seguintes parâmetros que permite a migração de máquinas vivas entre servidores.

```
(xend-relocation-server yes)
(xend-relocation-port 8002)
(xend-relocation-address '')
(xend-relocation-hosts-allow '')
```

A partir daí, o serviço de máquinas virtuais é reiniciado com o comando `“/etc/rc.d/xend restart”`.

Depois que o serviço de máquinas virtuais é reiniciado, deve-se verificar se a porta 8002, definida como a porta de comunicação para migração das máquinas virtuais, está aberta.

4.3.7 Outras configurações gerais no sistema

Quando da simples instalação do sistema o cluster não inicializa automaticamente. Isso se deve pelo fato dos arquivos de inicialização, localizado no diretório `/etc/rc.d` do sistema operacional dos servidores de virtualização, iniciarem de forma alfabética.

Um pré-requisito para a inicialização automática do sistema de virtualização, do cluster e do iSCSI é a ordem dessa inicialização, pois, se o OCFS2 inicializa antes do iSCSI, ele não localiza o dispositivo `/dev/sdb`, não monta o diretório do cluster e então, não funciona de forma automática, exigindo interação para iniciar.

Desta forma, se faz necessária a alteração do nome do arquivo “open-iscsi” para “niscsi”. Ou seja, deve possuir um nome que ocorre antes da letra o e depois da palavra network, pois também não pode iniciar antes da inicialização do serviço de rede.

Após isso, deve ser excluído da inicialização o arquivo open-iscsi e incluído o arquivo niscsi. Isso se faz com o comando `chkconfig -d open-iscsi` e a inclusão do novo arquivo de inicialização, com o comando `chkconfig -a niscsi`.

Essa configuração não está referenciada em nenhuma documentação, seja ela oficial ou não.

5 Testes

5.1 Teste de disponibilidade

No teste de disponibilidade, inicialmente, a máquina virtual é transferida de um servidor para o outro e não deve apresentar perdas de serviços ou, caso haja perda de serviço, MTTR baixo para o ambiente proposto.

Na análise, adotaram-se duas maneiras de testes distintas, conforme explicado a seguir.

5.1.1 Análise de tempo de indisponibilidade

Para se aferir o tempo de indisponibilidade do ambiente, caso seja necessária a migração da máquina virtual entre servidores, será executado um comando de ping e verificado quanto tempo esta máquina fica fora do ar como demonstrado no Plano de teste 1 tabela 4 abaixo. Serão adotadas duas referências para este teste, sendo elas, o tempo de indisponibilidade na migração da máquina virtual sem a opção de ocorrer com a máquina em execução e o simples boot em uma máquina física.

Tabela 4 - Plano de teste 1

Item	Tarefa	Resultado
1	Na máquina física	
1.1	Inicia máquina física cliente	Não se aplica
1.2	Acessa a interface de comando	Não se aplica
1.3	Executa o comando ping “ip” onde ip é o endereço ip da máquina virtual	Receber resposta de ping corretamente
1.4	Inicia o processo de reinício	1 – aumento da latência 2 – durante o reinício, ocorrer perda de pacotes
1.5	Reinício concluído	Receber resposta de ping corretamente
2	Na máquina virtual com migração simples	
2.1	Inicia máquina física cliente	Não se aplica
2.2	Acessa a interface de comando	Não se aplica
2.3	Executa o comando ping “ip” onde ip é o endereço ip da máquina virtual com migração simples	Receber resposta de ping corretamente
2.4	Inicia o processo de migração	1 – aumento da latência 2 – durante a transferência, ocorrer perda de pacotes
2.5	Migração concluída	Receber resposta de ping corretamente
3	Na máquina virtual com migração em tempo de execução – Live Migration	
3.1	Inicia máquina física cliente	Não se aplica
3.2	Acessa a interface de comando	Não se aplica
3.3	Executa o comando ping “ip” onde ip é o endereço ip da máquina virtual	Receber resposta de ping corretamente
3.4	Inicia o processo de migração	1 – aumento da latência 2 – durante a transferência, ocorrer perda de pacotes
3.5	Migração concluída	Receber resposta de ping corretamente

Este teste foi executado por dez vezes para se ter a média de execuções e ocorrências e o desvio padrão dessas.

Com o resultado dos testes, geramos a tabela 5 onde se pode ver que, no que tange a disponibilidade do ambiente, a máquina virtual que foi migrada em tempo de execução apresenta um tempo de estabilidade muito maior que as máquinas de referência.

Tabela 5 – Resultado do teste de indisponibilidade

Servidor / Forma de migração	Média de pacotes transmitidos	Média de pacotes recebidos	Média de pacotes perdidos	% de perda	Tempo médio de indisponibilidade	Desvio Padrão
Máquina Física	187,3	124,56	62,74	33,49	51,7 seg	7,2
Máquina Virtual / Migração simples	201,5	163,2	38,3	19%	38,7	6,5
Máquina Virtual / Live Migration	214,7	213,3	1	0,46%	1 seg	0

Isso se deve a forma com que a máquina é migrada, mantendo a funcionalidade enquanto os dados de memória são copiados de uma máquina para a outra. O ganho de estabilidade é visível, se levarmos em conta a máquina física, na qual foi feito um simples processo de boot.

5.1.2 Análise de latência

Para se analisar o aumento na latência durante a migração, será feita a transferência de um arquivo via rede de dados fast ethernet, via FTP durante o processo de migração do ambiente, com o procedimento demonstrado no plano de teste 2, tabela 6. Este teste visa verificar se a máquina pode ser considerada ativa

durante o processo de migração. Como referência será verificada a latência da transferência do mesmo arquivo em um ambiente virtualizado em funcionamento normal, ou seja, sem que seja feito o processo de migração.

Tabela 6 - Plano de teste 2

Item	Tarefa	Resultado
1	Na máquina virtual normal	
1.1	Inicia máquina física cliente	Não se aplica
1.2	Acessa a interface de comando	Não se aplica
1.3	Executa o comando ftp “ip” onde ip é o endereço ip da máquina virtual	Aparece solicitação de login e senha
1.4	Entra com usuário e senha	Aparece o prompt de comando ftp>
1.5	Executa o comando mget *.lzma	É solicitada confirmação do comando
1.6	Começa a receber o arquivo	
1.7	Conclui o recebimento do arquivo	Recebe o relatório de tempo de download e txa de transferência
1.8	Escreve o comando bye	Sai do aplicativo
2	Na máquina virtual com migração	
2.1	Inicia máquina física cliente	Não se aplica
2.2	Acessa a interface de comando	Não se aplica
2.3	Executa o comando ftp “ip” onde ip é o endereço ip da máquina virtual	Aparece solicitação de login e senha
2.4	Entra com usuário e senha	Aparece o prompt de comando ftp>
2.5	Executa o comando mget *.lzma	É solicitada confirmação do comando
2.6	Começa a receber o arquivo	
2.7	Inicia o processo de migração	- Aparece no computador que receberá o host migrado a máquina virtual. - A taxa de transferência cai
2.8	Conclui o processo de migração	A taxa de transferência volta ao normal
	Conclui o recebimento do arquivo	Recebe o relatório de tempo de download e txa de transferência
	Escreve o comando bye	Sai do aplicativo

Este teste foi executado por dez vezes para se ter a média de execuções e ocorrências e o desvio padrão dessas.

O arquivo para ser trafegado e levar um tempo suficiente para atender a migração do ambiente, foi o arquivo kde-i386.tar.lzma que possui tamanho de 102MB. Durante o processo de migração foram levantados alguns dados que estão na tabela 5.

Tabela 7 – Resultado do teste de latência

Equipamento	Tempo de download médio (s)	Taxa de transferência média (MB)	Desvio Padrão
Migração	87.369,39	0,84	0,08
Normal	48.160,43	1,97	0,12

O principal objetivo alcançado nesse teste foi verificar que a funcionalidade, mesmo necessitando de uma conexão persistente, permanece operacional, sem perda de função. Com isso já se garante a estabilidade do ambiente migrado mesmo em caso de necessidade de migração.

Outra observação importante é o aumento na latência, ou seja, no tempo de transferência de arquivos. Isso, em parte, se deve ao uso do mesmo canal de transferência de dados e de sincronismo entre os equipamentos, o que acaba causando uma concorrência pela banda de rede existente e assim, o aumento da latência na transmissão dos dados. Mas apesar disso, é visível que o tempo de transferência do arquivo aumenta consideravelmente.

A ferramenta utilizada nesse teste foi o aplicativo pure-ftp, que é um servidor de FTP gratuito e disponível dentre os pacotes padrão do open-suse.

5.2 Análise de desempenho

Para analisar a perda de desempenho gerada durante a migração de máquinas virtualizadas, irá ser executada, durante a migração da máquina virtual, a aplicação Apache Banchmark para verificar qual a perda de desempenho entre uma máquina virtual em execução e uma máquina virtual sendo migrada. O procedimento para este teste está descrito no plano de teste 3, tabela 8.

Tabela 8 - Plano de teste 3

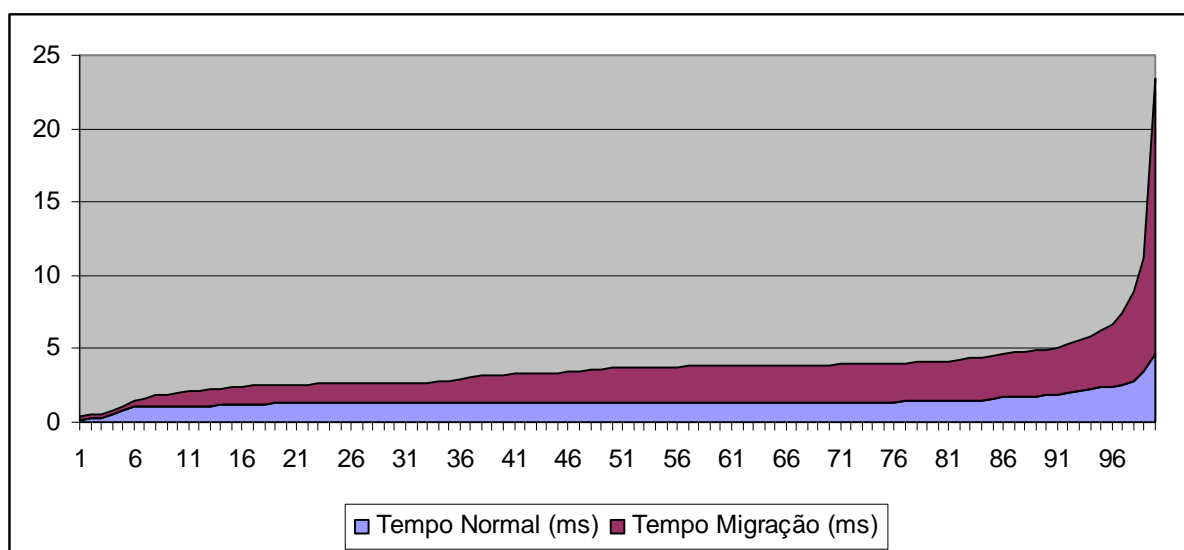
Item	Tarefa	Resultado
1	Na máquina virtual normal	
1.1	Inicia máquina virtual	Não se aplica
1.2	Acessa a interface de comando	Não se aplica
1.3	Executa o comando <code>ab -n 200000 -kc 10 http://ip/default.htm</code> onde ip é o endereço ip da máquina virtual	Inicia o processo de teste
1.4	Conclui o processo de teste	Aparece um relatório de como o teste se procedeu
2	Na máquina virtual normal	
2.1	Inicia máquina virtual	Não se aplica
2.2	Acessa a interface de comando	Não se aplica
2.3	Executa o comando <code>ab -n 200000 -kc 10 http://ip/default.htm</code> onde ip é o endereço ip da máquina virtual	Inicia o processo de teste
2.4	Inicia o processo de migração com o comando <code>xm migrate -l "host virtual" "nome do servidor de destino"</code>	A máquina virtual passa a aparecer no controlador de máquinas virtuais do endereço de destino parada - diminui a performance do benchmark
2.5	Conclui o processo de migração	A máquina virtual inicia no servidor de destino.
2.6	Conclui o processo de teste	Aparece um relatório de como o teste se procedeu

Para análise, foi utilizada a mesma máquina virtual e os testes ocorreram em momentos subsequentes, gerando os resultados descritos no Gráfico

1. Na máquina em que esta ocorrendo a migração, é visível que ocorre uma

lentidão na resposta, quanto maior e a quantidade de conexões simultâneas, chegando, quando se aproximam de 100.000 conexões, ao tempo máximo de 18,6 ms contra 4,7 ms na máquina virtual em execução normal. Porém, esta perda é esperada e se normaliza após a migração

Gráfico 1 – Tempo de Resposta por Conexões simultâneas



Verifica-se também, na tabela 9 que a capacidade interna da máquina virtual cai durante sua migração, ou seja, não é apenas o servidor que está promovendo a migração que tem sua capacidade reduzida, mas também a própria máquina virtual que está sendo migrada perde desempenho durante o processo de migração.

Tabela 9 – Resultado do teste de desempenho

	Unidade	Virtual	Migração
Tempo de Teste	Milissegundos	32,370	53,681
Requisições por segundo	Segundos	3089,27	1862,87
Tempo por requisição	Milissegundos	1,619	2,684
Taxa de transferência	KB/s	1224,29	738,26

6 Conclusão

Quanto à disponibilidade, não se obteve êxito na migração automatizada de máquinas virtuais, tendo em vista que esta tecnologia, para plataformas abertas, ainda está se firmando, sua instalação não foi estável e não foi possível realizar testes sobre ela. Porém, quando se trata de paradas programadas, o funcionamento da migração de máquinas vivas ocorreu com pleno êxito, possibilitando manutenções do ambiente físico sem a necessidade de parada de serviços e sistemas.

Quanto a Estabilidade, como visto nos testes desenvolvidos, no ambiente proposto houve o ganho da estabilidade independente do hardware que o sistema está instalado. Viu-se também que nos processos de migração, o tempo de indisponibilidade é próximo de zero, e a continuidade do serviço independe da disponibilidade dos servidores nos quais as máquinas virtuais estão instaladas. A perda de desempenho é mínima, mantendo o ambiente funcional e permitindo sua aplicação comercial.

Quanto a Simplicidade de instalação, a instalação de máquinas virtuais em um cluster de alta disponibilidade não é uma tarefa simples e nem trivial. Um ambiente com tamanha complexidade deve sempre ser analisado minuciosamente antes de se tomar a iniciativa de propor a sua instalação.

Para que o funcionamento ocorra de forma automática, customizações devem ser feitas nos scripts de inicialização do sistema, o que, normalmente, não é uma tarefa simples para qualquer tipo de usuário ou empresa.

Porém, é possível a criação deste ambiente, ficando a cargo dos administradores de rede a análise do conhecimento e capacidade de cada um de instalar este ambiente.

Quanto ao TCO desta solução. O custo financeiro da instalação é menor que a instalação de um ambiente físico e menor também que a instalação do ambiente virtualizado baseado em softwares proprietários como demonstrado na tabela 10.

Tabela 10 – Análise de TCO

	Software Livre	Software Proprietário
Custo do Software para solução	R\$ 0,00	R\$ 30.000,00
Custo de Hardware	R\$ 1.798,00	R\$ 1.798,00
Total	R\$ 1.798,00	R\$ 31.798,00

Com a estrutura proposta neste projeto, é possível a redução do TCO em mais de 90% e o ganho de disponibilidade do ambiente em mais de 50%.

6.1 Projetos Futuros

Às perspectivas de evolução deste projeto são:

- Desenvolvimento de uma interface de controle unificada para gerenciamento completo do cluster e das máquinas virtuais
- Aprimoramento da solução com base de dados de máquinas virtuais distribuída
- Aprimoramento da segurança da solução utilizando SSL no tráfego das máquinas virtuais

- Aprimoramento do microcódigo dos processadores residenciais para permitir virtualização completa em processadores AMD ou Intel.

Referências

BITTMAN, Thomas J. **Predicts 2004: Server Virtualization Evolves Rapidly**. USA, Gartner Inc. 2003

CAMBRIDGE, Universidade de. Computer Laboratory Systems Research Group – NetOS. <http://www.cl.cam.ac.uk/research/srg/netos/> Ultimo acesso em 01/06/2009

CLARK, Christopher; FRASER, Keir; HAND, Steven et al. **Live Migration of Virtual Machines**. University of Cambridge Computer Laboratory, USA e Department of Computer Science of University of Copenhagen, Denmark. [2005]

FAGUNDES, Eduardo Mayer. Gestão de contratos de SLA. www.efagundes.com. Ultimo acesso em 20/09/2008

MARSHALL, David; REYNOLDS, Wade A.; MCCRORY, Dave. **Advanced Server Virtualization: VMware® and Microsoft® Platforms in the Virtual Data Center**. Flórida, USA, Auerbach Publications 2006

MASHRAN, Sunil. **OCFS2 A Cluster File System for Linux. User guide for release 1.4**. Oracle Julho/2008 http://oss.oracle.com/projects/ocfs2/dist/documentation/v1.4/ocfs2-1_4-usersguide.pdf Ultimo acesso em 06/04/2009

PEREIRA, Roberto. **Alta Disponibilidade em Sistemas GNU/LINUX utilizando as ferramentas Drdb, Heartbeat e Mon**. Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras. Cap 4, p. 13-21. Minas Gerais, 2005.

ROBERTSON, Alan. The High Availability Linux Project. <http://www.linux-ha.org/>. Ultimo Acesso em 18/04/2008

WILIANS, David E; GARCIA, Juan. **Virtualization With XEN: The book that demystifies Xen**. Syngrees 2007.

RULE, David; DITTNE, Rogier. **The Best Damn Server Virtualization Book Period**. Everything you need to know about virtualization and leading virtual machine products

SATRAN, J; Meth, K; Sapuntizakis, V; Chadalapaka, M; Zeidner, E; 2004. Internet Small Computer Systems Interface (iSCSI). IETF 3720

Site do Guia Livre do Governo Federal. http://guialivre.governoeletronico.gov.br/mediawiki/index.php/P%C3%A1gina_principal. Ultimo Acesso em 22/09/2008

Site do Guia Livre do Governo Federal.
<http://guialivre.governoeletronico.gov.br/guiaonline/guiacluster/node44.php#SECTION06225000000000000000>. 2007. Último Acesso em 04/06/2009

SOARES, Rodolfo Rodrigues. **Utilização de Tecnologia Virtual, Máquinas e Redes Virtuais, em Data Center.** Trabalho de conclusão do Curso de Engenharia da Computação do UniCEUB. 2006

SZTOLTZ, Lisiane e outros. **Guia do Servidor Conectiva Linux.**
<http://www.dimap.ufrn.br/~aguiar/Livros/Conectiva9Server/book.html>. Último acesso em 14/04/2008

VMWARE; VMware Infrastructure 3. Data center management and optimization suite. 2006

Anexo I – Proposta comercial Star Company do Brasil

Brasília – DF, 28 de novembro de 2008.

Ao
ILMO. SR.
EDUARDO GONÇALVES MORICI

REF.: Cotação de Preços Nº. 135/2008

PROPOSTA COMERCIAL PROPOSTA/ORÇAMENTO

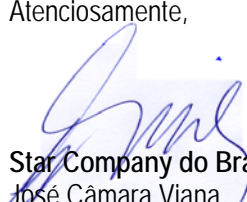
Objetivo: Venda de licenças do produto VMware VI3 (ESX 3.5) plataforma x86 para 2 licenças ESX Enterprise e 1 Virtual Center e Suporte Platinum de 3 anos.

Qtd	Produtos	Descrição	Valor Unitário	Valor Total
2	VMware Infrastructure Enterprise for 2 processors	SnS is required. Includes ESXi or ESX, VirtualCenter Agent, Virtual SMP, VMFS, Update Manager, Consolidated Backup, High Availability, VMotion/Storage VMotion, and DRS.	4.410,000	8.820,00
2	VMware VirtualCenter Server for VMware Infrastructure	SnS is required. Priced per VirtualCenter application instance	3.815,00	7.630,00
2	Platinum Support/Subscription VMware Infrastructure Enterprise for 2 Processors; additive licenses for 3 years	24 Hour Phone Support, 7 days a week	3.614,00	7.228,00
2	Platinum Support/Subscription VirtualCenter Server for VMware Infrastructure for 3 years	24 Hour Phone Support, 7 days a week	12.568,50	25.137,00
* TOTAL DO PROJETO SEM DESCONTO Para 2 Licenças ESX Enterprise, 1 Virtual Center e Suporte Platinum				30.000,00

Forma de pagamento: À VISTA

Validade desta Proposta: ATÉ DIA 29 DE DEZEMBRO DE 2008

Atenciosamente,



Star Company do Brasil - Soluções integradas em TI ®
José Câmara Viana
Consultor de Negócios e Projetos
RG 442810 SSP DF

Anexo II – Nota Fiscal de aquisição de equipamentos

09/05/09 COMPANHIA BRASILEIRA DE DISTRIBUIÇÃO **NOTA FISCAL - MODELO 1** Nº 270079

extra S/MAS, S/N TRECHO 3 LOTE 5
GUARÁ - BRASÍLIA - DF
Tel.: (061) 2101-8200

☒ SAÍDA ☐ ENTRADA 270080

1ª VIA DESTINATÁRIO/REMETENTE

NATUREZA DA OPERAÇÃO: **Venda a consumidor** CUF: **5929** INSCRIÇÃO ESTADUAL DO SUBSTITUTO TRIBUTÁRIO: **047.508.411/1184-56** INSCRIÇÃO ESTADUAL: **07.304.787.036-82** DATA ABRE P. EMISSÃO: **09/05/09**

DESTINATÁRIO/REMETENTE: **EDUARDO MORICI GONCALVES** CNPJ/CNP: **000070338167153** DATA DA EMISSÃO: **09/05/09**

ENDEREÇO: _____ BAIRRO/DISTRITO: _____ CEP: _____ DATA DA ÚLTIMA ENTRADA: **09/05/09**

MUNICÍPIO: _____ FONE-FAX: _____ UF: _____ INSCRIÇÃO ESTADUAL: _____ HORA DA SAÍDA: **20:22**

DADOS DO PRODUTO

CODIGO PRODUTO	DESCRIÇÃO DOS PRODUTOS	DT. TRIB.	UNID.	QDNT.	VALOR UNITÁRIO	VALOR TOTAL	ALÍQUOTA
8815974	IMBR 26B160HD LNX	11		2	898,000	1.796,00	700
8837250	MONITOR ADC 15LCD 519SW	11		2	1,000	2,00	700
Subtotal TI : 1.798,00							

PRAZO DE TROCA 72 HORAS
GARANTIMOS A TROCA DO PRODUTO, COM EMBALAGEM E ACESSÓRIOS ORIGINAIS EM ATÉ 72 HORAS APÓS O RECEBIMENTO DA MERCADORIA. Após este prazo, dirija-se a assistência técnica autorizada.

S A I D
PREVENÇÃO DE PERDA
Em 09/05/09
EXTRA 1366

<< Imposto desta nota foi recolhido através do Cupom 246206 do ECF 049 >>

CÁLCULO DO IMPOSTO

VALOR DO IMPOSTO	VALOR DO ICMS	BASE DE CÁLCULO DO ICMS-SUBSTITUIÇÃO	VALOR DO ICMS-SUBSTITUIÇÃO	VALOR TOTAL DOS PRODUTOS
1.798,00	125,86			1.798,00
VALOR DO FRETE	VALOR DO SEGURO	OUTRAS DESPESAS ACESSÓRIAS	VALOR TOTAL DO I.P.	VALOR TOTAL DA NOTA
				1.798,00

TRANSPORTADOR / VOLUMES TRANSPORTADOS

NOME: _____ RAZÃO SOCIAL: **O PRÓPRIO** FICHA POR EXCITA: ☐ PLACA DO VEÍCULO: _____ Nº: _____ CNPJ/CNP: _____

ENDEREÇO: _____ BAIRRO: _____ UF: _____ INSCRIÇÃO ESTADUAL: _____

QUANTIDADE: _____ ESPÉCIE: _____ MARCA: _____ NÚMERO: _____ PESO BRUTO: _____ PESO LÍQUIDO: _____

DADOS ADICIONAIS

RESERVADO AO FISCO

Nº DE CONTROLE DO FORMULÁRIO: **225319**

Assinatura: *Recebo em 09/05/2009*
Cia Brasileira de Distribuição
extra 1366

Nº IE: _____
At. do CAVE Administração do Gvra
151 ou 5012-1651